U.S.A. $2.50
CANADA $3.00

ATARI PRINTER SURVEY

Antic™

The ATARI® Resource

JANUARY 1984 VOLUME 2, NUMBER 10

- **SPECIAL PRINTER ISSUE**
- **Program a 1984 Calendar**
- **Animate by Page Flipping**
- **Blossoming of Atari Computer Art**
- **Atari Screen Dump**
- **'Snowbird' Game**
- **Atari Computers & Epson Printers**
- **Rotating Printer Fonts**



0 74470 12728 01

ATARI—Official Home Computer of the 1984 Olympics

Have You Read The Reviews On Our New AtariWriter™?



Atari's New Word Processor.

Try it, be your own critic. Use new AtariWriter for all your family correspondence, school papers, committee reports, mailings.

Our advanced ROM-based cartridge technology means you can use it on any ATARI Home Computer (even 16K). It also lets you choose between cassette and disk storage systems.

Easy to learn and to use. "Yet," says ANALOG Magazine, "as your uses for word processing grow, (you'll employ) its more advanced features."

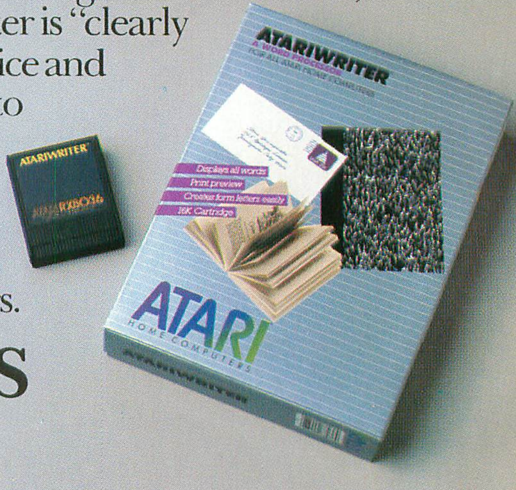
Menu driven, with helpful prompts. One special feature: you can correct as you write, instantly, without switching back and forth between Create and Edit modes.

You can move entire paragraphs. Search and replace. Even restore text you've just deleted, with our memory buffer.

You can change format instantly, and preview on-screen at full printer width. Print extra copies for your records.

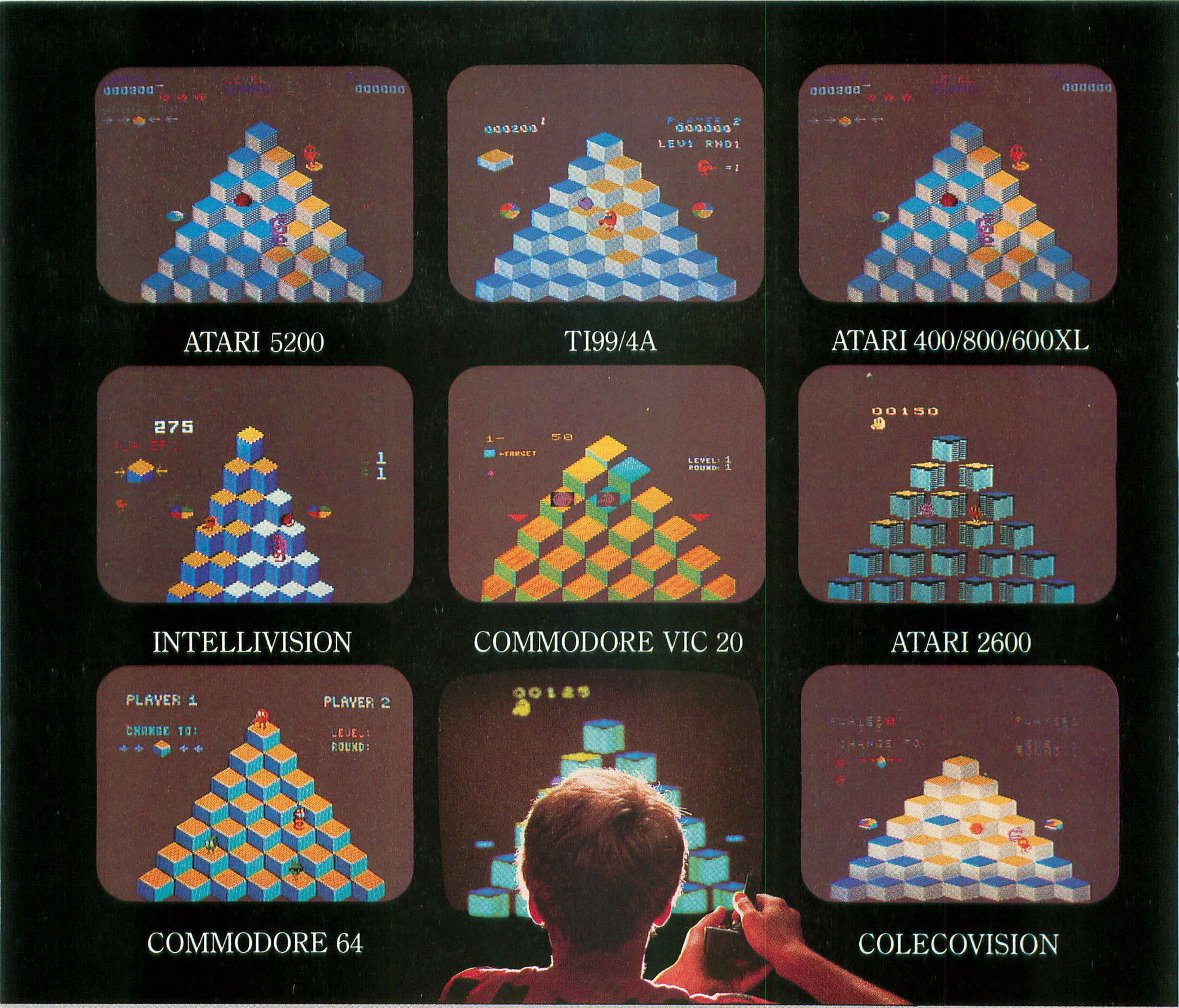
As ANTIC Magazine concluded, new AtariWriter is "clearly superior on price and performance to Text Wizard and the ATARI Word Processor."

At Atari dealers.

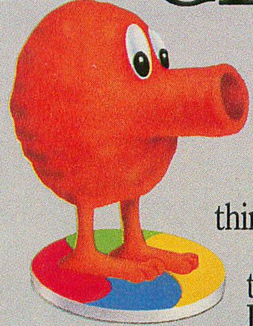


ATARI® HOME COMPUTERS

A Warner Communications Company.



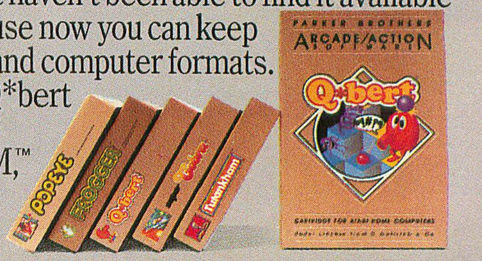
HOW TO GET Q*BERT™ OUT OF YOUR SYSTEM.



If you've been wanting to play Q*bert, but haven't been able to find it available for your home system, your time has come. Because now you can keep things hopping with any of these popular home video and computer formats.

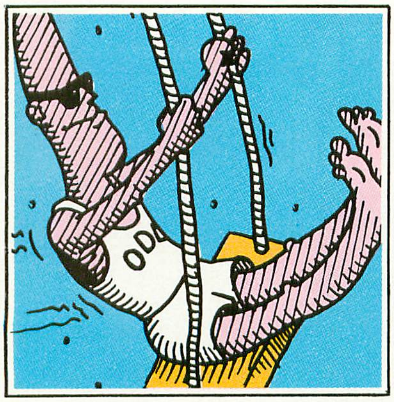
Get going to your nearest video store and get Q*bert today. And while you're there, check out Parker Brothers' POPEYE,® FROGGER,™ TUTANKHAM,™ and SUPER COBRA.™ All the great Arcade Action® games, now in all the great home formats.

PARKER BROTHERS

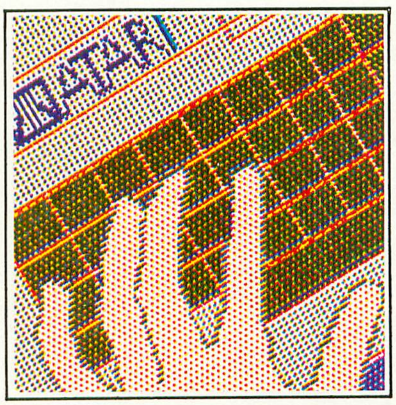


Antic™

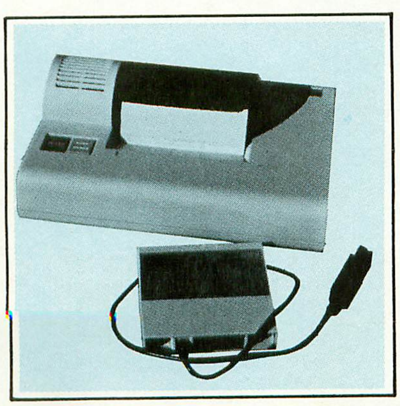
The ATARI Resource JANUARY 1984 VOLUME 2, NUMBER 10



page 14



page 34



page 53

FEATURES

DEPARTMENTS

We just made owning an Atari computer a lot more logical.



Introducing the Rana 1000 disk drive. It's a whole new game for Atari computers.



This two digit LED readout displays a code that tells you everything you need to know.

This beeping button tells you your write protect feature is keeping your information safe.

The remaining buttons beep when touched, and provide readouts on density storage, error status, and drive number.

This button beeps when you touch it, and the LED readout tells you what track you're on.

When Rana Systems introduced the Elite Series of Apple® compatible disk drives, we didn't know what a tremendous impact they would make. It turned out to be a line so outstanding in performance, styling, capacity, and price, that it instantaneously made us a major force in the market. Well, needless to say, the response was so great that we were forced to create the same highly advanced disk drive for Atari.® A disk drive that when coupled with Atari's computer, could perform everything from accounting, financial planning, and stock charting, to word processing, business management, and letting you write your own programs. Plus, we made it simple enough for a child to use, for learning anything from the alphabet to a foreign language.

Working with a diskette versus playing with a cassette.

Let's face it. The only reason Atari made a cassette option to their computer was to make it affordable. But now you don't have to settle for less. Because now you can get a diskette for your Atari computer which outperforms their cassette and costs ⅓ less than their disk drive. With Atari's cassette you only get half the functions of a computer compared to what our floppy disk can give you. Their cassette is not only limited in the software available, but it also takes 20 times longer to get the information you need. And Rana's disk drive offers twice the storage capacity of either their cassette or disk drive.

Why even stylewise our new low profile design not only looks 100 times more spectacular, but it occupies 3 times less space. And our new Rana 1000 also gives you a piece of its mind every time you use it, because our disk drive gives you information as well as takes it. And we think that says a lot.

The disk drive that has all the answers.

Rana offers you a myriad of features Atari couldn't even conceive of. Like five electronic functions on the front panel that actually beep and give you a LED readout when touched. Our disk drive tells you what track you're on, and what density and how much information you're storing. It lets you switch from a single density of 90,000 letters to a double density of 180,000 letters, on a single diskette. And, we have a write protect feature which protects your diskette from being erased. In fact, no other disk drive can offer you that.

As you can see, it was easy to build a disk drive superior to Atari's. Because for every reason you buy a disk drive, Rana has superior technology.

The Rana 1000 disk drive. It brings your Atari computer to a higher level of sophistication for a price one third lower than Atari's. So your choice shouldn't even be a matter of logic.

Just common sense.

RanaSystems *Always a step ahead of the originals.*

21300 Superior Street, Chatsworth, CA 91311 213-709-5484. For dealer information call toll free: 1-800-421-2207. In California only call: 1-800-262-1221. Source Number: TCT-654

Available at all participating Computerland stores and other fine computer dealers.

® Apple is a registered trademark of Apple Computer, Inc. ® Atari is a registered trademark of Atari, Inc., a Warner Communications Company. See us at the West Coast Computer Show. Circle 44

Antic
The ATARI Resource

Publisher
James Capparell
Managing Editor
Robert DeWitt
Assistant Editor
Christopher Rauber
Technical Copy Editor
David Duberman
Editorial Assistant
Caitlin Morgan
Editorial Clerk
Arla Ertz
Contributing Editors
Carl Evans
Ken Harms
John & Mary Harrison
David & Sandy Small
Jerry White
Art Director
Marni Tapscott
Production Artist
Kyle Bogertman
Contributing Illustrator
Beatrice Benjamin
Cover Illustration
John Mattos
Circulation Manager
Les Torok
Subscriptions
Julianna Hoffman
Advertising/Production Coordinator
Linda Tapscott
Advertising Sales
Steve Randall
Accounting
V.J. Briggs

ANTIC is an independent periodical not affiliated in any way with Atari, Inc. ATARI is a trademark of Atari, Inc. All references to Atari products are trademarked and should be so noted.

January 1984
Volume 2, Number 10
ANTIC—The ATARI Resource, is published twelve times per year by ANTIC Publishing. Editorial offices are located at 524 Second Street, San Francisco, CA 94107. Telephone is (415) 864-0886. ISSN 0745-2527. Second Class Postage paid at San Francisco, California and additional mailing offices. POSTMASTER: Send address change to **ANTIC**, 524 Second Street, San Francisco, CA 94107.

Editorial submissions should include program listing on disk or cassette, and text file on media and paper if text was prepared with a word processor. Media will be returned if self-addressed stamped mailer is supplied. **ANTIC** assumes no responsibility for unsolicited editorial material.

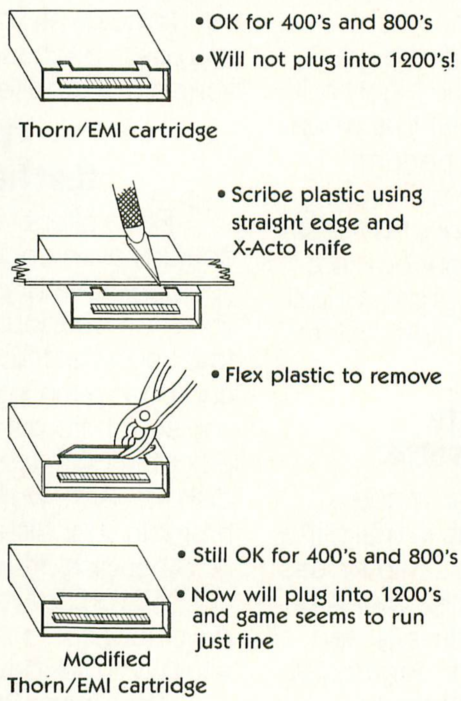
Editorial and Subscription Office (415) 864-0886
Circulation Office (415) 863-2340
Advertising Sales Office (415) 661-3400
Accounting Office (415) 552-4485

I/O BOARD

CARTRIDGE CHAOS

I recently purchased Thorn/EMI's Jumbo Jet Pilot game because the salesman assured me that it would work on my Atari 1200XL. I soon discovered, however, that the cartridge would not even plug into my computer! After investigating, I determined that the cartridge's case was preventing it from loading properly. The following illustrations show how I modified my Thorn/EMI cartridge. It plugs in now, and seems to play well (though I haven't mastered the game yet).

Bruce Whiteside
Woodridge, IL



TOGGLE TROUBLE

I've suffered through a lot of frustration at the hands of the Atari logo key — the key that toggles the inverse-video mode on and off. I've come up with a simple solution to this problem that I'd like to share with other ANTIC readers. I simply inserted a routine at the beginning of the vertical blank that checks to see if the Atari logo key has been pressed. If it has, my routine presses it again (internally, of course), thus turning the inverse-video mode off. The program follows:

```
10 For I=1541 TO 1553: READ X:
   POKE I,X: NEXT I
20 DATA 174,176,128,208,5,162,0,
   142,182,2,76,5,233
30 POKE 549,6
```

Tim Miltz
Indiana, PA

A SCREEN DUMP, PLEASE

At the suggestion of the Atari Hotline, I'm writing to ANTIC for information that I've been unable to find anywhere else. I'm looking for a printer-driver or SAVE routine, in either machine language (6502) or BASIC, that will enable me to transfer screen graphics from an Atari 800 with a disk drive to an Epson (or similar) printer with bit-mapped graphics. Can you help me?

James I. Taylor
Nashua, NJ

Yes. See Jerry White's "Screen Dump" in this issue. —ANTIC ED

THE BEST OF ANTIC

I think your magazine is wonderful, especially the arcade-style games that appear in each issue! If possible, I'd like to obtain reprints of the following games: Chicken (ANTIC, April 1982), Death Star (ANTIC, June 1982), Speed Demon and Frog (ANTIC, October/November 1982). Thank you for your help.

Mark Bing
Smyrna, GA

You're in luck! The games you mentioned are no longer available in magazine form, because the stocks of our first four issues have been completely depleted. But three of the games you've requested, along with Bats and six brand-new, arcade-style games, are now available in a new book from ANTIC, "The Best of ANTIC, Volume 1." This collection of the most popular programs and feature articles from our first year of publication is available at

continued on page 10

many local bookstores and computer stores, or you can order it directly from ANTIC. For only $12.95, this ANTIC anthology should keep you playing games for a long time to come.

—ANTIC ED

BACK ISSUE BLUES

I would like to obtain ANTIC issues 1, 2, 3 and 4 (Volume 1), and these issues are no longer available through the publisher. Is there anybody out there who has copies of any of these issues and is willing to part with them? I'll pay up to $4.00 per issue for copies that are in reasonably good condition.

Kent R. Redding
Sandy, UT

We're always glad to hear that our back issues are still popular. If anyone is interested in Kent's offer, write to him in care of ANTIC and we'll see to it that he gets your letter. Note that much of that material appears in our book, "The Best of ANTIC, Volume 1."

—ANTIC ED

QUESTION-MARK QUESTION

Is there any way to blank out the question mark that results from an input statement (other than positioning a blank space where it would be)?

Jeff Dreyer
Tacoma, WA

See several solutions by Harvey Bernstein in the May 1983 issue (ANTIC, I/O Board, page 6). —ANTIC ED

REACH OUT!

This past summer, my family and I visited friends in another part of the country. During the trip I stopped in at an Atari store in the hope of trading programs with someone and perhaps making a friend in the process. It worked! I now have a "computer pal," and I've discovered that people in other states know about different publications, magazines and software sources than we do in Ohio. I enjoy getting new programs and new ideas through the mail.

With this in mind, I wonder if there are other ANTIC readers who would be interested in finding computer pals. Perhaps ANTIC could start a new section for people to write in to for this purpose.

Randy Raymond
Dunkirk, OH

ANTIC will consider starting such a service if there is sufficient reader interest in a feasible plan. Please let us know whether you want such a service and how you would like to see it run. Would you like a "Computer Pal" classified section in the magazine? Would you like to have ANTIC randomly reroute your "Computer Pal" letters to other readers who are interested in making new computer friends? Send your comments or suggestions to PALS, care of ANTIC, 524 Second St., San Francisco, CA 94107. —ANTIC ED

SAVE IT FIRST

If I type in a machine language program, what do I have to do before I press RUN? I read somewhere that you have to save your program twice. Is that true?

A. Batek
Don Mills, Ontario,
Canada

A machine language program, if presented in a magazine to be typed in, is usually written in a form called assembly language. It must be typed in, then assembled, and then you must specify the address at which the program is to start running.

Most of the programs we print, however, are written in BASIC. After typing one in, you should SAVE at least one copy on tape or disk before typing in RUN and pressing [RETURN]. The reason for this is that RUNning a program can lock up your computer. If this happens, and your program was not SAVEd, you have to type in the entire program again. Saving two copies of the program is a good idea, because it gives you an extra backup in case something happens to one of them. —ANTIC ED

APPLAUSE, APPLAUSE

I've been a regular reader of your magazine for about a year — though not a subscriber — and I want to thank you for your help. ANTIC got me through the "beginner" stage, and I'm grateful for it. The only reason I'm not a subscriber is that most of the other computer magazines I've read publish issues occasionally that can only be described as "dogs." But you've been going strong for over a year now with not even a bark!

ANTIC is terrific — not only the feature articles and the games, but the reviews as well. Your review of Miner 2049er (ANTIC, page 87, February/March 1983) convinced me to buy the game, and I've yet to regret the decision.

Ellen Keyt
Mt. Clemens, MI

A TYPEWRITER/PRINTER?

Would you recommend buying one of the new electronic typewriters that can also interface with a computer and function as a printer? I'm particularly interested in the Brother Correctronic CE50. How does it compare to other printers in quality and price? Is it difficult to use with an Atari 800?

Edwin H. Reveche
Baldwin Park, CA

The Brother Correctronic CE50 electronic typewriter is one of many new typewriters that come with an optional computer interface. Virtually all manufacturers of typewriters are making models with this feature; many, like Brother, are designed specifically to be used as a computer printer. Its print mechanism (without the keyboard) is used in several printer models.

Hooking up the Brother to the Atari is simple — you can get the typewriter,

continued on page 12

The ultimate current event.



Edison, the kinetic android, leads a frustrating life.

All he wants to do is build his circuit boards and go with the flow. But things keep getting in the way.

Nohms—a negative influence—bug him constantly. They're harmless, but only from a distance.

Flash, the lightning dolt, disconnects everything in his path. Which can be frustrating after a hard day on the circuit.

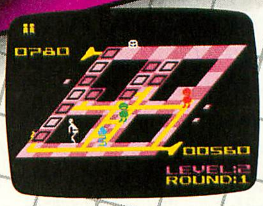
And the cunning Killerwatt is out to fry poor Edison's brains. But our hero simply solders on.

Compute's Gazette calls Juice! "a knockout...the graphics are top-notch and the play action is fun for five minutes as well as five months."

Try Juice! yourself. You'll get a charge out of it.

Developed by Arti Haroutunian
Available for your Commodore 64™
and Atari Home Computer.

JUICE!



TRONIX™

8295 South La Cienega Blvd.
Inglewood, CA 90301 (213) 215-0529

I/O BOARD

interface box and Atari 850 cable as a package and simply plug them together. The interface box is an attractive unit with a standard Centronics-style plug. It automatically goes on when the printer does. Any word-processor that allows you to use a "dumb" printer should work with this setup — AtariWriter and Letter Perfect both work well.

The Brother provides high-quality daisywheel print. Other advantages include a one-year warranty, local service and easy availability. However, like all daisywheel printers, it is very slow, printing at 12 CPS. Daisywheels are expensive ($25) and the only easily-obtained ribbon is a carbon-film, non-reusable one ($6) that produces high-quality print, but only about 20 pages worth.

The typewriter is available at discount stores for $399 or less, and the interface costs about $200. This combination is an outstanding buy if you need high-quality print, need to use a typewriter occasionally, and don't mind the slow speed. (This information was submitted to us by Michael Dunn, president of Atari Computer Enthusiasts of Eugene, Oregon, and a long-time friend of ANTIC.) —ANTIC ED

HELP!

TROUBLE WITH TYPO

Your article "TYPO" (ANTIC, page 35, August 1982; reprinted in April 1983, page 116) explains how the program works, but it doesn't go quite far enough. For example, what do CODE and LENGTH mean, and how do they correspond to each other? Computer magazines sometimes take it for granted that their readers understand everything about the inner workings of a program — and that's just not so. An explanation in layman's terms should accompany each article. This would be a great help to the beginning programmer.

David Stroup
Marietta, PA

We probably do sometimes take too much for granted. Please keep in mind, though, that it's difficult to produce a magazine that will be of interest to a wide variety of readers with very different levels of computing experience. However, we do try to explain our terms and concepts in language that the beginner can understand.

Typo is a very valuable program, but it has confused a number of our readers. Careful reading of the text reveals that "length" is "the number of bytes encountered by TYPO within the line number range shown." The "code" is "a checksum of the 'length' bytes in that same range." A new monospaced version of TYPO should make its appearance in the February 1984 issue. —ANTIC ED

LIST-ASSISTER ASSISTER

I'm very pleased with the "List Assister" program that ran in your July 1983 issue (ANTIC, Systems Guide, page 93) and its September 1983 upgrade (ANTIC, HELP!, page 10). Now I can use my daisywheel printer to list programs including *all* the necessary characters. John Shreffler, a friend of mine, has made some additional enhancements that I think are worthwhile. They ensure that the right-hand margin is consistent (John and I noticed that some programs, such as AMODEM4, did not print out consistently all the way across the page). The rewritten lines are as follows (as written by List Assister, of course):

```
365 IF (II+DL-1)>(LL+7) THEN ?
    #4; CHR$(155);B$(5);:II =
    LL+7:L=L+1:LL=LL+65*
    (T=18)+120*(T=15)
370 ? #4;"[";L$;"]";:II=II+DL-1:
    NEXT I
430 IF A=128 THEN ? #4;"<,>";:
    II=II+2:NEXT I
460 IF A=0 THEN ? #4;"[,]";:II=II
    +2:NEXT I
490 IF A=253 THEN ? #4;"<;>":
    II=II+2:NEXT I
500 A=A-128:? #4;"|";CHR$(A);:
    II=II+1
540 TRAP 730:? #4;"|";:? #4:
    GOTO 590
```

Keep up the good work!

Richard Reaser
Los Angeles, CA

WRONG NOTE!

There is an error in the listing that accompanies "Pick A Chord" in ANTIC Volume 2 Number 8 (November 1983) that prevents the program from functioning correctly. In the second line of line 80, in which the string MENU2$ is defined, an extra space should be inserted after the first character, which is the numeral 7. This will move the first m in min7b5dim7 to directly beneath the M in MENU2$ in the line above.

AIR RAID 2000

In the program Air Raid 2000 (ANTIC, November 1983), line 980 lists out to four lines. By using the abbreviation SE. for SETCOLOR, however, this line can be entered within the Atari BASIC limit of 114 characters (120) if you POKE 82,0).

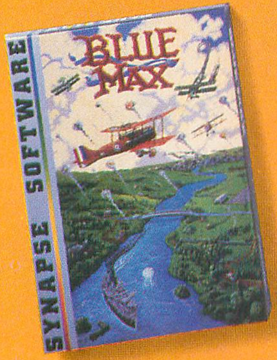
Also, characters that appear to be inverse periods in lines 850 and 860 are actually inverse commas.

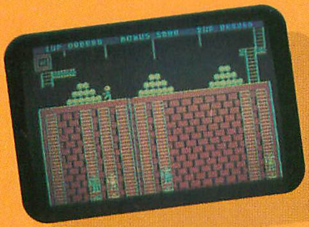
SYNAPSE EXCITEMENT



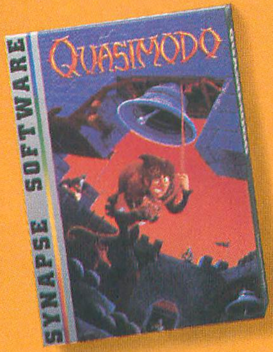


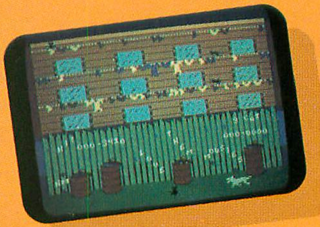
On patrol
Out of the sun comes your RAF biplane, loaded down with a deadly cargo of bombs and bullets. But watch out for the anti-aircraft guns and the enemy fighters—a hit could mean a tricky landing for repairs and ammo. **BLUE MAX.***



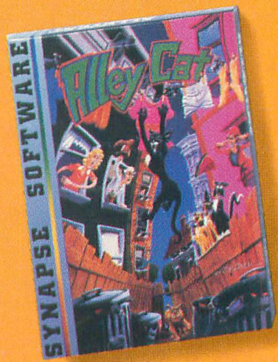


Stop the ringing!
Quasimodo knows who stole the crown jewels. He even knows where they are, but the soldiers just won't leave him alone. Help Quasi dodge the guards and return the jewels to their rightful owner. **QUASIMODO.***

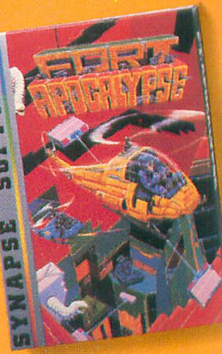




Catastrophic!
Freddy the Cat is busy all day long, getting into trouble every time he jumps in a window. If he's not dodging bottles, bones and shoes he's chasing mice or avoiding that big bully Bowser von Spike! **ALLEY CAT.***



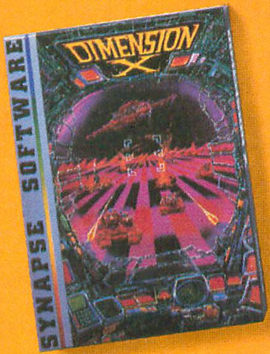
Take the controls
Your helicopter mission—capture vital fuel and weapons, free the enslaved masses, and finally destroy the fortress itself. Will you triumph or be crushed by the fiendish Kraalthan lords? **FORT APOCALYPSE.***



FOR YOUR ATARI!

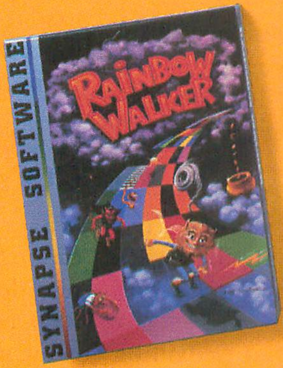


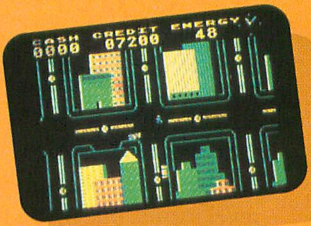
Surrounded!
There's nothing like a Rigillian invasion fleet to spoil your day! You just have time to activate the shield envelope and weapons systems before the enemy ships fill the scanner screen. Good luck—you'll need it! **DIMENSION X.***





Color the world
Someone stole all the colors of the rainbow. It's up to Cedrick to put on his magic shoes and hop from square to square putting back the color. But watch out for the Borbuls, the Devils, the Gruebirds and the big holes in the rainbow! **RAINBOW WALKER.***



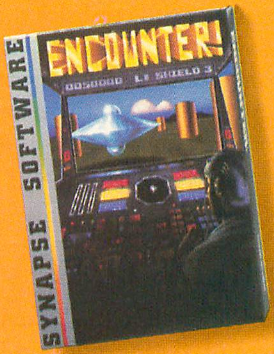


Big Apple bites back
A typical Midwestern tourist takes his first big bite out of the Big Apple. Little did you realize just how much trouble you could get into in broad daylight! **NEW YORK CITY.***





Can you survive?
Alone on a distant planet, you stumble into an ancient combat training course that really plays for keeps. Can you survive the deadly drones and sneaky saucers? **ENCOUNTER.***



Write for a FREE catalog and Elite Club information.

Synapse games are also available on disk and cassette for the Commodore, Apple and IBM home computers.

Name ______________________

Address ______________________

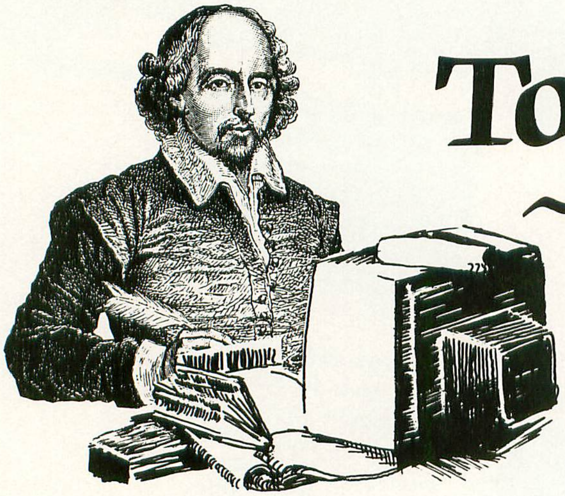
City ______________________

State __________ Zip __________

Computer ______________________ A

synapse SOFTWARE

5221 Central Avenue, Richmond, CA 94804 · 415/527-7751



To IOCB ~or~ Not To IOCB

Open communications with I/O Control Blocks

by MARK GRICE

Input/Output Control Blocks, commonly referred to as IOCB's, are probably the most versatile utilities the programmer has for extracting and storing information. Quite a few novice programmers shy away from them because they seem to be cumbersome and hard to understand, but they're really quite simple.

IT'S CHILD'S PLAY

When I think about IOCB's, I often recall a TV show I used to watch as a kid. It was called "Davy and Goliath." The main character, Davy, often communicated with his friends by using two Campbell's soup cans that were tied together with a string. I don't know if you've ever tried this method of communication, but I have. And in the process, I learned a number of interesting things about it:

1) Before you can talk, you need to give an end to each of the parties involved.

2) Only one person can talk at a time. Because of this, you must decide whether you will talk or listen before the communication starts.

3) Before someone else can use the cans to talk, the first two users must end their communication.

These may not appear to be startling revelations, but if you understand these three simple rules you will have no trouble using IOCB's.

THE ELEMENTS OF AN IOCB

An IOCB always starts with an OPEN command, and looks something like this: OPEN #2,8,0,"C:". The elements that make up this sample command work as follows:

- OPEN #2, — This sets up a line of communication (think of it as one of the cans, along with the string) that originates in the computer. This is called line #2.
- 8, — This tells us that we will be outputting (or, as in the tin-can example, talking). A four in this position means we're inputting; a six means that we're inputting a disk directory; a nine means that we're appending a file that's already in existence; and a twelve can mean that we're either inputting or outputting.
- 0, — This tells us nothing, in most cases. It's just there. I know alot of you are screaming, "Why?" but I'm going to hedge on this anyway. Suffice it to say that there is a purpose for this part of the command, but that it's used so seldom for this purpose that it's not worth going into. Over 98 percent of the time, this number will be zero.
- "C:" — This is where the IOCB stops. It designates the other end of the IOCB channel — who gets to use the can at the opposite end of the string, in a manner of speaking. (In disk operation, we'd also need a filename here, so the command would look something like this: OPEN #2,8,0,"D:FILENAME". In disk I/O, by the way, opening with the number eight will wipe out whatever was in the file previously, while opening with the number nine will cause the program to start writing on the next available sector on the disk, regardless of how much room is left in the previous sector.)

A LINE OF COMMUNICATION

What we have now is a line of communication from the computer to the cassette recorder (for the purposes of this article, we'll assume that a cassette recorder is being used for output). Again, this line is called LINE #2. The computer is doing the talking; the cassette is listening. A short subroutine using this line might look like this:

```
20 PRINT "ENTER THE
   STUDENT'S SCORE"
30 INPUT SCORE
40 PRINT "THE STUDENT'S
   SCORE IS:",SCORE
50 OPEN #2,8,0,"C:"
60 PRINT #2;SCORE
70 END
```

Line 50 has already been explained. Lines 20 and 30 are pretty straightforward, and I don't think they require any further explanation. In line 40, we print the score in the regular fashion, to the screen. Line 60, however, prints the score to IOCB #2. This time, the infor-

Mark Grice, 22, is a self-taught programmer and computer tutor. He has trained more than one hundred students to program, and two of his former students are now professional software authors. Mark is also interested in robotics, and has built and programmed an Atari-controlled robot.

mation doesn't go to the screen — it goes to the cassette, instead. Why? Because we've told the computer to PRINT #2.

Going back to our example of the tin cans and string, the scenario would look like this:

Davy is holding the tin can. On the other end of the string is his friend Bob. With Davy is another friend, Jane. She is planning a picnic, and wants to invite Bob. So Jane asks Davy:

"What time should we hold the picnic?" (20 PRINT "WHAT TIME IS THE PICNIC?")

Davy thinks about it for a moment (30 INPUT ANSWER)

"Oh, about 3:30," he says. (40 PRINT "OH ABOUT,":ANSWER)

Then Jane says, "Better tell Bob." So Davy picks up the can (50 OPEN #2,8,0,"C:") and says "Hey Bob, the picnic is going to be at 3:30." (60 PRINT #2;ANSWER)

I hope I haven't insulted anyone's intelligence with my examples, but honestly, that's all there is to it! Anything connected to your computer can have an IOCB attached to it. This includes printers, cassettes, disk drives, the keyboard, or even a TV (ever heard of a light pen?).

A FEW QUICK RULES

You can number your IOCB's using any number between one and seven (i.e., line #1, line #2, line #3), but you should avoid using the number 6, because it is sometimes used by the Operating System (OS). For example, have you ever seen a line like the following?

```
10 GR. 2: PRINT #6; "HELLO"
```

In this example, the OS opens an IOCB to the graphics screen rather than the text screen.

When you're finished with an IOCB, close it. An END statement will close all IOCB's. This ensures that all of your transmissions will be received by the output device (in technical terms, it empties the buffer). If you don't close your IOCB, there is an excellent chance that some of your data will be lost. To close an IOCB without ending the program, use the CLOSE statement (e.g. CLOSE #2).

Finally, after an IOCB is closed you can open it again to another device, but you *cannot* have the same IOCB opened to two devices simultaneously (just as Davy couldn't have had one string attached to three cans). For more information on IOCB's, read your Reference manual.

A FINAL EXAMPLE

Here's one last example of how you can use IOCB's effectively. The situation is as follows: you have to create a menu program that is to be used by someone who is not terribly bright. There are two things he just can't seem to remember: 1) That when you ask for a number, he can't enter a letter; and 2) Where the [RETURN] key is located on the keyboard.

The first program you submitted looked like this:

```
10 ? "PLEASE MAKE SELECTION"
20 ? "1) CHOOSE ONE"
30 ? "2) CHOOSE TWO"
40 ? "3) CHOOSE THREE"
50 ? "ENTER NUMBER PLEASE"
60 INPUT CHOICE
70 IF CHOICE = 1 THEN
   GOTO 100
80 IF CHOICE = 2 THEN
   GOTO 200
90 IF CHOICE = 3 THEN
   GOTO 300
```

However, you quickly discovered that this would never work, because every time your user ran the program, he either started a letter or forgot to press [RETURN]. An IOCB could help you tremendously:

```
10 ? "PLEASE MAKE SELECTION:"
20 ? "1) CHOOSE ONE"
30 ? "2) CHOOSE TWO"
40 ? "3) CHOOSE THREE"
50 ? ENTER NUMBER PLEASE"
60 OPEN #1,4,0,"K:": GET
   #1,CHOICE
70 IF CHOICE = 49 THEN
   GOTO 100
80 IF CHOICE = 50 THEN
   GOTO 200
90 IF CHOICE = 51 THEN
   GOTO 300
95 GOTO 10
```

This method works like a charm! As soon as the user touches a key, the program takes off. No matter what the user does, he can only select a number. The explanation for this magic is that in line 60 we've opened up an IOCB for input (because the second number is a four) from the keyboard. The computer then treats this input in the same way that it treats any input: It simply waits until a key is pressed.

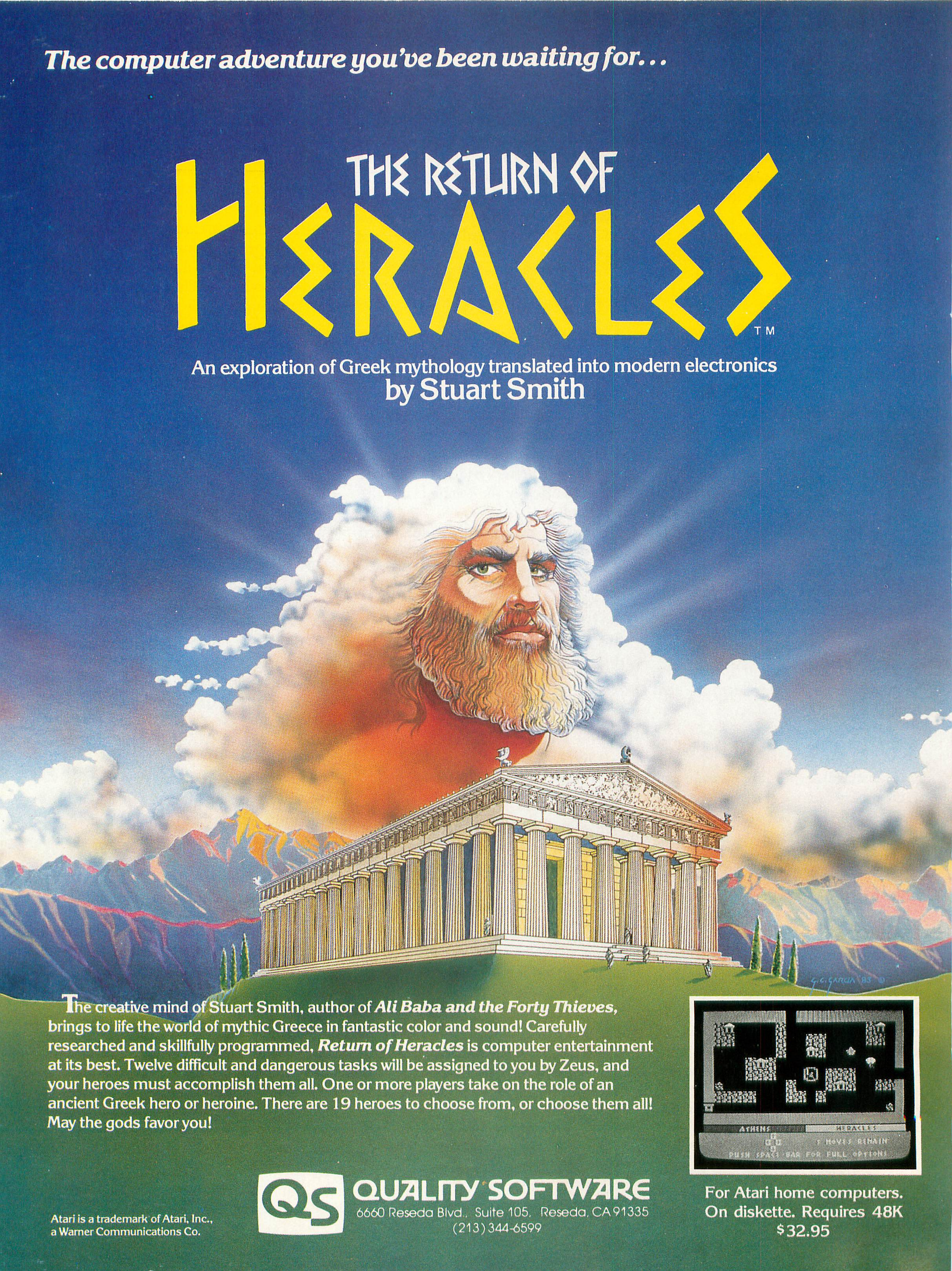
However, unlike the standard INPUT command, which must wait for a press of [RETURN] to tell it that an option has been selected, the GET command in this program takes effect after a single byte of information has been received. In this case, that single byte is always the ATASCII value of whatever key has been pressed by our hypothetical user.

By the way, using the IOCB GET command is not the same as PEEKing location 764, which I discussed in an earlier article (ANTIC, "I/O and You," page 16, July 1983). Location 764 gives you the interrupt value, which is the value you get before the OS gets its hot little hands on it. The IOCB GET command gets the value only after it is sent to the OS. This is why, in this case, the IOCB GET command returns the ATASCII value of any key that is pressed.

As a result, when we use this method we know for certain that we will end up with a number, rather than a letter, no matter what happens. So we can say goodbye to Error 8. Incidentally, if we want to print the selection that our user has made, we can do so easily by adding the following line:

```
65 PRINT CHR$(CHOICE)
```

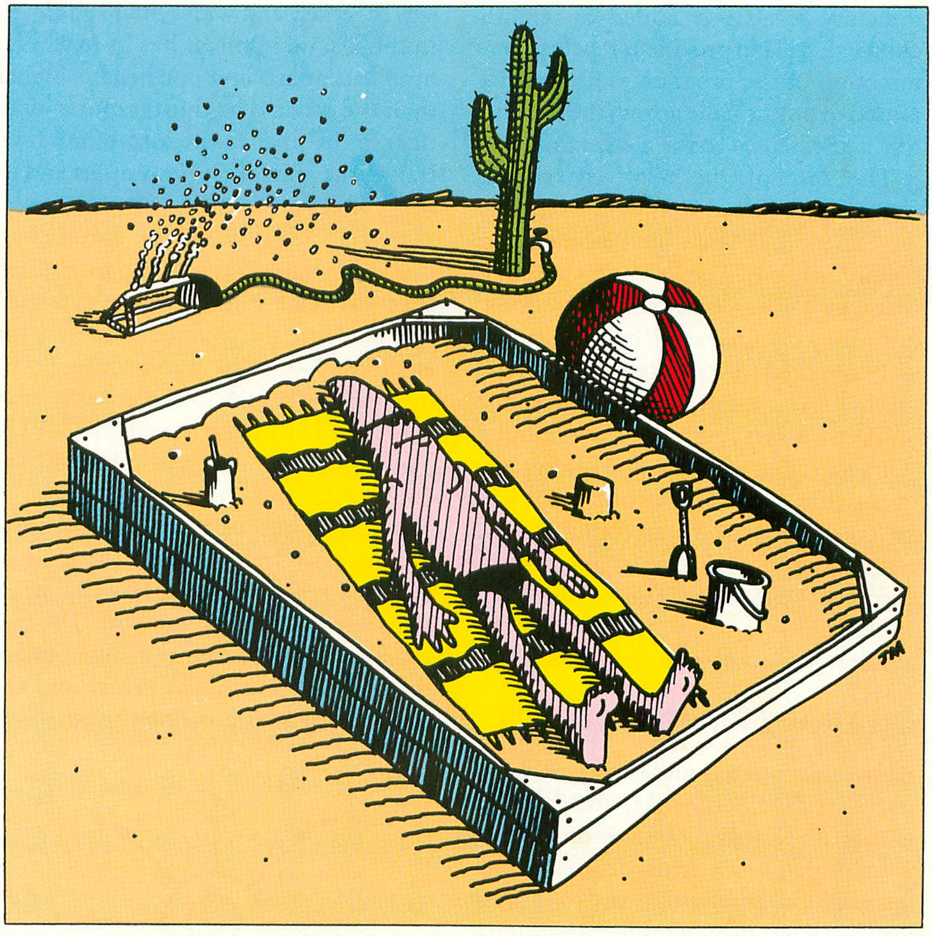
In conclusion, if you want constant input that you can "grab on the fly," use location 764. If, on the other hand, you want simple input and you want the computer to wait for it, use an IOCB. Only after exhausting these two possibilities should you use an INPUT statement! Try them. I think you'll find that these two options will help make your programs much more "user-friendly."


The computer adventure you've been waiting for...
THE RETURN OF
HERACLES™
An exploration of Greek mythology translated into modern electronics
by Stuart Smith
The creative mind of Stuart Smith, author of Ali Baba and the Forty Thieves, brings to life the world of mythic Greece in fantastic color and sound! Carefully researched and skillfully programmed, Return of Heracles is computer entertainment at its best. Twelve difficult and dangerous tasks will be assigned to you by Zeus, and your heroes must accomplish them all. One or more players take on the role of an ancient Greek hero or heroine. There are 19 heroes to choose from, or choose them all! May the gods favor you!
QS
QUALITY SOFTWARE
6660 Reseda Blvd., Suite 105, Reseda, CA 91335
(213) 344-6599
For Atari home computers.
On diskette. Requires 48K
$32.95
Atari is a trademark of Atari, Inc., a Warner Communications Co.

ODD MAN AT PLAY

Transforming a concept into a game

by JOHN and MARY HARRISON



Last month we developed the custom character set that is the backbone of **Odd Man Out**, an educational program we designed to develop visual discrimination skills in preschoolers. This month, we will examine the subject of *play action*, the interaction between computer and user that transforms a concept into a game, as part of our study of how to write educational programs for preschool children.

THE BASICS OF PLAY ACTION

No matter how well conceived an idea seems to be, it is not effective unless it is used. Because of this, one of the most important facets to consider in designing educational software is play action. In developing Odd Man Out, we considered the following elements:

- varying background colors with difficulty level
- choice of graphics mode
- making the joystick responsive for young children
- adding an animated character to use in selecting proper response
- random selection of displayed objects
- immediate reward for correct response
- immediate indication of incorrect responses (without being critical)
- graphic reward based on performance

Some of these items are quite simple and require little or no explanation, but it is well worth the time it takes to examine the implementation of the more complex elements.

IT'S THE SIMPLE THINGS THAT COUNT

The pre-school audience lacks the sophistication of older and more computer-aware age groups. The three or four year old doesn't care if the animation is jerky or the graphics are simple — if he can gain mastery over the computer and make it perform for him. While this should not be used as an excuse for shoddy work, it's true that elaborate or highly-detailed graphics displays are rarely appreciated by the preschooler. Often it is the simple, subtle touches that make a program for youngsters easy to use and interesting to play.

By varying the background color with each difficulty level, we can immediately cue both parent and child to the level of

difficulty being attempted. This is simply accomplished. In Odd Man Out, the variable CHOICE designates which of the seven levels has been chosen. Since the Atari is capable of displaying sixteen basic colors, we multiplied CHOICE by two and used the SETCOLOR command to change the background color. Color selection is arbitrary, but we preferred to use the even-numbered color choices.

To make the menu appealing, we used Graphics Mode 1 which contains large characters and thus makes the menu easy to read. An arrow points to the current selection. In addition to the seven difficulty levels, we added an eighth choice, EXIT, to provide for a clean termination of the program. The joystick is used to position the arrow; the fire button is pressed to make a selection.

The last of the relatively simple elements we included in Odd Man Out was a joystick routine designed for children's needs. The joystick is a switch that can sense eight distinct directions, but young children often lack the motor skills required to accurately position the joystick. As a result, when it checks for an up or down response, for example, our program checks for and allows any of the three joystick positions that can be considered to be pointing down, rather than allowing only one correct choice (see Figure 1). This minimizes a young child's anxiety by widening the window for acceptable responses.

CHARACTER ANIMATION

A more obvious technique we used to improve the Play Action of Odd Man Out is that of character animation. While animation is an extremely complex element of programming, and one that usually results in crude and jerky motion in BASIC, our implementation of animation in Odd Man Out is fairly simple and yields smooth motion.

Because this technique only works in Graphics Mode 0, we used the text window at the bottom of the Graphics Mode 2 display as the background for our animation. The Graphics-0 characters are small, so we designed an object three-characters-wide and two-characters-high. We wanted to be able to move the object both to the left and the right, and to keep its implementation simple. As a result, a two-headed animal seemed appropriate (our apologies to Dr. Doolittle). A child can position our "pushme-pullyou" beneath the object he or she wishes to choose without facing the distraction of seeing an animal that appears to walk backwards.

We started this process by attempting to visualize how such an animal would move. We decided that four "frames" would be required to describe the motion of the pushme-pullyou. In the first frame, the animal stands with its front and hind legs together, and with both heads level. In the second frame, it has its front and hind legs apart, and one head is higher than the other. The third frame is identical to the first. In the fourth and final frame, the animal's legs are apart and its other head is raised.

Having identified these frames, we drew the figures on graph paper. Then the individual characters were extracted and read into the computer (as were the custom characters we discussed last month). Next, the complete sequence was stored in a string variable, PMPY$. This sequence is shown in lines 1380 and 1400 of Listing 1.

These lines appear to be quite obscure, so let's look at them more closely. Remember it takes six characters to define the pushme-pullyou. The first three characters define the upper half of the animal. The down arrow and the three left arrows position the cursor so

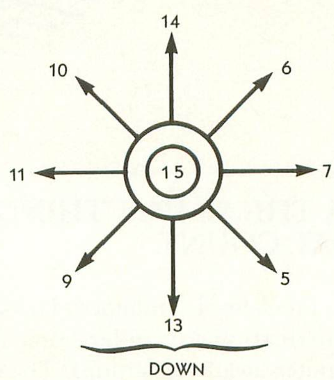


Figure 1
Odd Man Out allows for more than one correct choice of direction on the joystick.

that the bottom half of the pushme-pullyou is drawn beneath the upper half. This completes the first frame of the sequence. The rest of PMPY$ contains the remaining three frames of the animated sequence.

To prevent the danger of having bits and pieces of the pushme-pullyou strewn randomly about the screen, a mechanism is needed that erases those portions of the screen that are no longer occupied by the animal. This is accomplished by means of the variable ERASE$. Thus, we print ERASE$, the frame, and then ERASE$ again. This allows the animal to move to either the left or the right without leaving a trail of abandoned parts behind.

RANDOM DISPLAYS

Nothing makes a game more boring than being able to foresee what sequence of problems or obstacles will be displayed. We have attempted to avoid this pitfall in Odd Man Out by using the random-number feature of Atari BASIC. We use this random feature in three ways. First, we randomly select the object that will be displayed three times in a particular problem. Second, we randomly

continued on next page

John and Mary Harrison are parents, teachers and Atari hobbyists. Mary teaches math and computer science at the high school level. John holds an M.S. in computer science and develops educational software. They coordinate the Education Department for ANTIC.

"Odd Man At Play" is the third installment of a series designed to teach the basics of writing educational software. The series is based on a program called Odd Man Out. Part of the program listing appeared in the December issue; the listing and its explanation will be completed in February. As a result, those portions of the program that appeared in the December issue and that appear in this issue do not constitute a complete program. The system requirements of the completed program will be either 24K RAM (cassette) or 32K RAM (disk). —ANTIC ED

choose the object that will be displayed only once (the odd man out). Third, we randomly select the position that will contain the odd object. These steps ensure that there is no bias in our program regarding either the objects to be displayed or the location of the odd object in each set of objects.

EXPLANATIONS OF THIS MONTH'S PROGRAM SEGMENT

This month's program segment consists of the following major sections:

(1) Lines 140–560: declare variables and read in characters for the pushme-pullyou
(2) Lines 880–1300: select difficulty level
(3) Lines 1320–1420: define frame sequence for character animation
(4) Lines 1500–1860: create level 1, 2, and 3 displays
(5) Lines 1880–2200: create level 4 and 5 displays
(6) Lines 2220–2480: create level 6 and 7 displays
(7) Lines 2500–2860: joystick input and pushme-pullyou move routines
(8) Lines 4180–4420: data that defines the pushme-pullyou

The first section requires little explanation. The variable FRAMES is the number of animation frames used (in this case four). FRAMSZE holds the length of a particular frame. In our case, ten characters are required per frame (six for the pushme-pullyou, along with four cursor-positioning characters used to properly align the object).

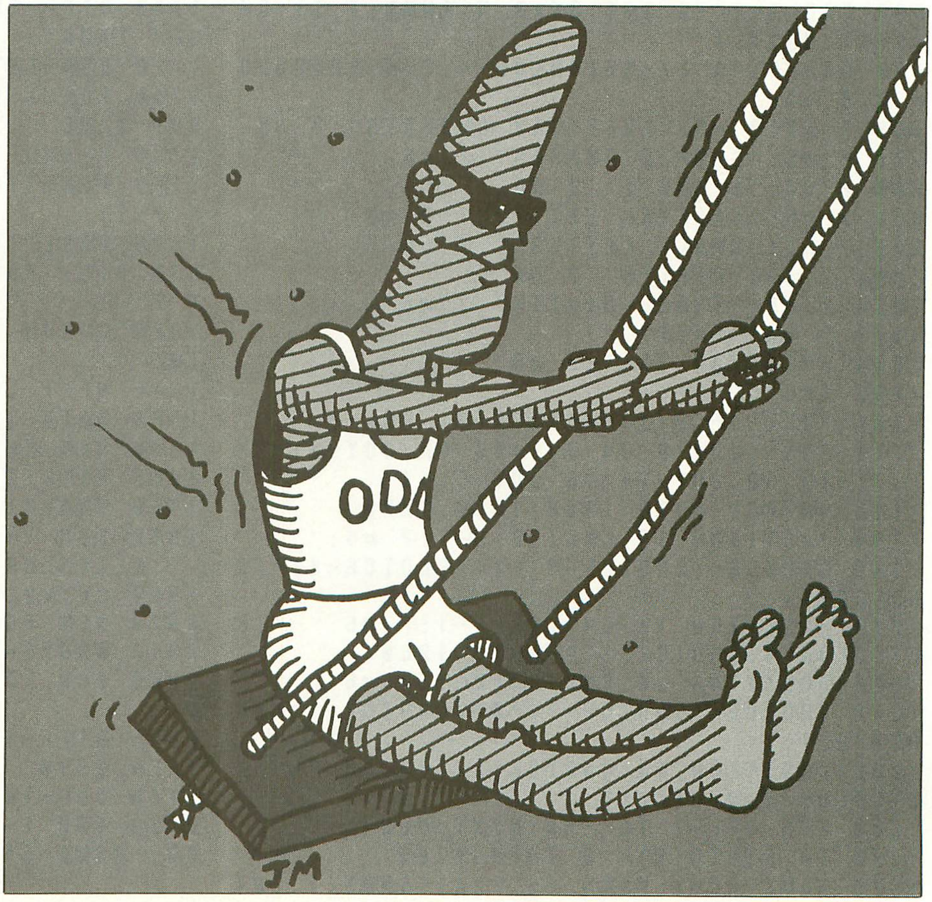
The next section of the code is also fairly straightforward. Lines 880–1080 present the menu in Graphics Mode 1 and set CHOICE to one. We then reset the Attract Mode Flag (POKE AF,0). This keeps the computer from automatically beginning the attract mode, since the keyboard is not used. Next, the joystick position is read. POS is then used to determine if the UP or DOWN motion on the menu is desired. POS is set to two if DOWN is desired or three if UP is indicated. The remainder of this section adjusts CHOICE to the appropriate value, changes the background color, moves the pointer, and checks to see if the fire button has been pressed.

The third section of the code copies the custom characters for the pushme-pullyou into the character set, and sets up the character-animation strings PMPY$ and ERASE$. Lines 1500–1860 replace some of the lines used last month with the proper code for levels 1, 2, and 3. This code is more interesting and requires some explanation.

Since Odd Man Out consists of ten problems, a FOR/NEXT loop is used to control problem setup. SAME selects the one object (out of eight possibilities) that is to be displayed three times in a particular problem. DIFF selects one of the remaining seven objects to be the odd object. ODD chooses the location (0 to 3) of the object chosen by DIFF. P points to the first character in the object to be displayed. The screen is printed and control is passed to the joystick-input routine at line 2500. Lines 1880–2480 are similar, except that they allow the standard Atari characters to be displayed as objects. The only modified characters are the answer box and the pushme-pullyou.

One line that may be confusing is 1940. To reduce the amount of work required, this line copies the lower-case letters into the first half of the character set at the locations of the capital letters (a = >A), rather than copying the pushme-pullyou, answer box, and other miscellaneous characters into the second half of the standard character set.

Lines 2500–2860 are both the heart of Odd Man Out and a place where Play Action becomes a serious consideration. POKE 752,1 turns off the cursor. Locations 656 and 657 hold the row and column, respectively, of the cursor in the Graphics window. The pushme-pullyou is displayed on the left-hand edge of the screen, and the joystick is checked for input. If the value of JOY is greater than seven, the joystick could not have been moved to the right, so control is passed to line 2660. If JOY is greater than 11,



this means that the joystick was not moved to the left, so control is passed to line 2800. If JOY <= 11 or <= 7, POS is updated and the character-animation sequence in lines 2700–2780 is activated.

The character-animation sequence displays the four frames required for the pushme-pullyou to take a step. By placing ERASE$ both before and after FRAME$, the left or right-hand edge of the pushme-pullyou that had been previously displayed will be erased. Lines 2740–2780 create the "clip-clop" sound effects of the walking character. The remainder of the section checks to see if the fire button has been pressed and if the pushme-pullyou is under one of the answer boxes. If it is, the next problem is displayed. Next month we'll add the right and wrong answer routines that need to be activated at this point.

PROGRAM ADDITIONS

It is now time to add this new section of code to the section presented in last month's column. Combining these two listings will provide you with the skeleton of Odd Man Out. However, there is still no indication in the program of which responses are correct or incorrect, or of an overall evaluation of a child's performance. These topics will be examined next month.

To combine the two listings, first type in Listing 1 and LIST it to disk or cassette tape. Then load last month's listing, ENTER the new listing, and save the resulting program to cassette or disk for future use.

```
10 REM *****************************
20 REM *                           *
30 REM *        ODD MAN OUT        *
40 REM *            BY             *
50 REM *   JOHN AND MARY HARRISON  *
60 REM *                           *
70 REM *            FOR            *
80 REM *      ANTIC MAGAZINE       *
90 REM *                           *
95 REM *****************************
140 DIM ANM$(104),NAME$(20),PMPY$(48),FRAME$(12),ERASE$(12),DISP$(20)
180 AF=77
320 FRAMES=4:FRMSZE=10:POKE AF,0
560 FOR I=1 TO 104:READ X:ANM$(I,I)=CHR$(X):NEXT I
880 GRAPHICS 17:SETCOLOR 4,2,8:SETCOLOR 0,0,14
900 ? #6:? #6;"EVERYDAY OBJECTS":? #6
920 ? #6;"SIMPLE SHAPES":? #6
940 ? #6;"E AND [":? #6
960 ? #6;"CAPITAL LETTERS":? #6
980 ? #6;"SMALL LETTERS":? #6
1000 ? #6;"DIGITS":? #6
1020 ? #6;"THREE DIGITS":? #6
1040 ? #6;"EXIT"
1060 POSITION 18,1:? #6;"<"
1080 CHOICE=1
1100 POKE AF,0:JOY=STICK(0):POS=0
1120 POS=(JOY=9 OR JOY=13 OR JOY=5)+2*(JOY=10 OR JOY=14 OR JOY=6)+1
1140 ON POS GOTO 1300,1160,1220
1160 POSITION 18,CHOICE*2-1:? #6;" "
1180 CHOICE=(CHOICE=8)*1+(CHOICE<8)*(CHOICE+1)
1200 POSITION 18,CHOICE*2-1:? #6;"<":SETCOLOR 4,CHOICE*2,8:GOTO 1280
1220 POSITION 18,CHOICE*2-1:? #6;" "
1240 CHOICE=(CHOICE=1)*8+(CHOICE>1)*(CHOICE-1)
1260 POSITION 18,CHOICE*2-1:? #6;"<":SETCOLOR 4,CHOICE*2,8
1280 FOR DEL=1 TO 100:NEXT DEL
1300 IF STRIG(0)<>0 THEN 1100
1320 RAM$=ROM$:RAM$(25,128)=ANM$(1,104)
1340 RAM$(473,488)=BOX$
1380 PMPY$="#$%↓←←←&'()*%↓←←←+,-"
1400 PMPY$(21)="#$%↓←←←&'(#./↓←←←+,-"
1420 ERASE$=" ←↑ ←↑ "
1440 ON CHOICE GOTO 1480,1500,1520,1960,1940,2240,2240,4140
1500 RAM$(145,208)=GEO$(1,64):RAM$(257,448)=GEO$(65,256):GOTO 1540
1520 RAM$(145,208)=E$(1,64):RAM$(257,448)=E$(65,256)
1640 WRONG=0
1660 FOR LOOP=1 TO 10
1680 SAME=INT(8*RND(0))
1700 DIFF=INT(8*RND(0)):IF DIFF=SAME THEN 1700
1720 ODD=INT(4*RND(0))
1740 FOR I=0 TO 3:P=(I=ODD)*DIFF+(I<>ODD)*SAME
1760 P=INDEX(P)+32
1780 POSITION I*4+3,5:? #6;CHR$(P);CHR$(P+1)
1800 POSITION I*4+3,6:? #6;CHR$(P+2);CHR$(P+3):POSITION I*4+3,8:? #6;"[":NEXT I
1820 GOSUB 2500:REM JUMP TO INPUT ROUTINE
1840 NEXT LOOP
1860 GOTO 3100
1880 REM SET UP FOR LEVELS IV, V
1940 RAM$(265,472)=RAM$(777,984)
1960 GRAPHICS 2:POKE 756,GRTOP
1980 FOR I=0 TO 3:SETCOLOR I,0,14:NEXT I:SETCOLOR 4,CHOICE*2,8
2000 SETCOLOR 2,CHOICE*2,8
2020 REM POKE 16,112:POKE 53774,112
2040 WRONG=0
2060 FOR LOOP=1 TO 10:SAME=INT(26*RND(0))
2080 DIFF=INT(26*RND(0)):IF SAME=DIFF THEN 2080
2100 ODD=INT(4*RND(0))
2120 FOR I=0 TO 3:P=(I=ODD)*DIFF+(I<>ODD)*SAME+65
```

continued on page 24

ODD MAN AT PLAY continued from page 22

```
2140 POSITION I*4+3,6:? #6;CHR$(P):POSITION I*4+3,8:? #6;"[":NEXT I
2160 GOSUB 2500:REM JUMP TO INPUT ROUTINE
2180 NEXT LOOP
2200 GOTO 3100
2220 REM SET UP FOR LEVELS VI, VII
2240 GRAPHICS 2:POKE 756,GRTOP:GOTO 2260
2260 FOR I=0 TO 3:SETCOLOR I,0,14:NEXT I:SETCOLOR 4,CHOICE*2,8
2280 SETCOLOR 2,CHOICE*2,8
2300 REM POKE 16,112:POKE 53774,112
2320 WRONG=0:M=10:N=0:IF CHOICE=7 THEN M=900:N=100
2340 FOR LOOP=1 TO 10:SAME=INT(M*RND(0))+N
2360 DIFF=INT(M*RND(0))+N:IF SAME=DIFF THEN 2360
2380 ODD=INT(4*RND(0))
2400 FOR I=0 TO 3:P=(I=ODD)*DIFF+(I<>ODD)*SAME
2420 POSITION I*4+3,6:? #6;P:POSITION I*4+3+(CHOICE=7),8:? #6;"[":NEXT I
2440 GOSUB 2500:REM JUMP TO INPUT ROUTINE
2460 NEXT LOOP
2480 GOTO 3100
2500 REM JOYSTICK MOVE AND INPUT ROUTINE
2520 POKE 752,1:FLAG=0
2540 POS=3:POKE 656,0:POKE 657,POS+1
2560 ? PMPY$(1,10)
2580 JOY=STICK(0):POKE AF,0
2600 IF JOY>7 THEN 2660:REM JOYSTICK RIGHT ROUTINE
2620 POS=POS+1:IF POS>34 THEN ? "⇱":POS=3
2640 GOTO 2700
2660 IF JOY>11 THEN 2800:REM JOYSTICK LEFT ROUTINE
2680 POS=POS-1:IF POS<4 THEN ? "⇱":POS=34
2700 FOR I=1 TO FRAMES:FRAME$=PMPY$(I*FRMSZE-(FRMSZE-1),I*FRMSZE)
2720 POKE 656,2:POKE 657,POS:? ERASE$;FRAME$;ERASE$;
2740 IF I/2<>INT(I/2) THEN SOUND 1,4,0,14
2760 IF I/2=INT(I/2) THEN SOUND 1,4,16,14
2780 SOUND 1,0,0,0:NEXT I
2800 IF STRIG(0)=1 THEN 2580
2820 RE=(POS-5-(CHOICE=7)*2)/8:LE=(POS-4-(CHOICE=7)*2)/8
2840 IF RE<>INT(RE) AND LE<>INT(LE) THEN 2580
2860 RETURN
4180 DATA 0,8,24,124,252,255,31,31
4200 DATA 0,0,0,0,0,129,255,255
4220 DATA 0,16,24,62,63,255,248,248
4240 DATA 31,15,7,3,3,3,3,3
4260 DATA 255,255,129,0,0,0,0,0
4280 DATA 248,240,224,192,192,192,192,192
4300 DATA 8,60,124,252,28,28,31,31
4320 DATA 0,0,0,0,0,0,255,255
4340 DATA 31,15,7,7,13,24,48,48
4360 DATA 255,255,129,0,129,195,102,102
4380 DATA 248,240,224,224,176,24,12,12
4400 DATA 0,0,0,0,0,128,255,255
4420 DATA 16,60,62,63,56,56,248,248
```

TYPO TABLE

Variable checksum = 1285912

Line num range			Code	Length
10	-	180	ZV	426
320	-	1060	FM	495
1080	-	1260	YF	527
1280	-	1660	QL	505
1680	-	1940	OO	468
1960	-	2180	BR	488
2200	-	2420	DX	551
2440	-	2660	QE	389
2680	-	4200	WE	511
4220	-	4420	XA	327

TRIVIA-QUESTION QUIZ

A quizzical way to learn

by GEORGE STONE

Are you having trouble learning your French vocabulary words? Or those historical dates you have to memorize by next Friday? Are you a sports, TV, or movie fan who loves to share trivia questions with your friends? Or are you a teacher with a constant need for question and short-answer drills?

Here is a skeleton program for the Atari that you can use in all of these situations. It displays a scoreboard at the top of the screen, with the user's name, the number of problems attempted, the number of correct answers, and the percentage of correct answers.

The questions are displayed below, and the user types in short-answer responses. A sequence of up or down sound notes is followed by a "correct" or "incorrect" message for each answer.

Once the skeleton program is loaded, all that is necessary to prepare your version is to add a set of DATA statements and a number in line 170. Line 170 says:

FOR SENT = 1 TO 3

Change the three to match the number of DATA statements you want to use. You can program your DATA statements at any free location you wish (line 1000 is a good place to start).

The DATA statements should be written as follows:

1000 DATA WHAT IS THE QUESTION?,ANSWER

That is, the line number, followed by the question, a comma, and the correct answer. Remember to use capital letters if you want the answers to be upper case, and to use a comma between the question and the answer.

Remember also to save the skeleton program before entering the DATA so that it can be used for other purposes again. Then save the completed program under a new filename. You can use the skeleton program to create a complete collection of educational drill or trivia programs.

You can enter as many questions as you like, and make the programs as long as you like, but you must be sure to make line 170 agree with the number of questions.

```
10 DIM R$(1):DIM WORD$(100):DIM SENT$(
100)
20 DIM ANS$(100):DIM NAME$(25)
30 X=0:Y=0
40 GRAPHICS 0:SETCOLOR 2,5,8:SETCOLOR
1,0,0:POKE 752,1
50 POSITION 2,8:? "TYPE THE USER'S NAM
E PLEASE.":? "THEN PRESS RETURN.":?
60 INPUT NAME$
70 ? CHR$(125)
80 POSITION 2,1:? NAME$
90 POSITION 2,3:? "NUMBER OF QUESTIONS
    ";X
100 POSITION 2,5:? "NUMBER CORRECT
      ";Y
110 POSITION 2,7:? "PERCENT CORRECT
 "
120 POSITION 2,0:? "__________________
________"
130 POSITION 2,8:? "__________________
________"
140 DIM BL$(40)
150 BL$=" ":BL$(40)=BL$:BL$(2)=BL$
160 REM FOR DLY=1 TO 200:NEXT DLY
170 FOR SENT=1 TO 3
180 POSITION 2,12:? "QUESTION ";SENT
190 PRINT
200 READ SENT$,WORD$
210 ? SENT$
220 ? :? "NOW TYPE THE CORRECT ANSWER.
"
230 ? :INPUT ANS$
240 IF ANS$=WORD$ THEN 350
250 SOUND 0,29,10,10:FOR DLY=1 TO 10:N
EXT DLY
260 SOUND 0,45,10,10:FOR DLY=1 TO 10:N
EXT DLY
270 SOUND 0,0,0,0
280 POSITION 2,20:? "THAT IS NOT CORRE
CT."
290 FOR DLY=1 TO 250:NEXT DLY
300 GOSUB 570
310 POSITION 2,12:? "THE ANSWER TO"
```

continued on next page

```
320 ? :? SENT$:?
330 ? "IS":? :? WORD$
340 GOTO 410
350 SOUND 0,35,10,10:FOR DLY=1 TO 10:N
EXT DLY
360 SOUND 0,31,10,10:FOR DLY=1 TO 10:N
EXT DLY
370 SOUND 0,29,10,10:FOR DLY=1 TO 10:N
EXT DLY
380 SOUND 0,0,0,0
390 POSITION 2,20:? "THAT IS CORRECT."
400 Y=Y+1:POSITION 25,5:? Y
410 X=X+1
420 POSITION 25,3:? X
430 S=INT((Y/X)*100)
440 POSITION 25,7:? "    [←][←][←]";S
450 POSITION 2,22:? "PRESS RETURN FOR
QUESTION ";SENT+1;
460 INPUT R$
470 GOSUB 570
480 NEXT SENT
490 POSITION 2,12
500 IF S=100 THEN ? "FANTASTIC! YOU DI
D GREAT!"
510 IF S<100 AND S>89 THEN ? "VERY GOO
D! THAT'S A GOOD SCORE!"
520 IF S<90 AND S>79 THEN ? "PRETTY GO
OD. YOU DID OK!"
530 IF S<80 AND S>69 THEN ? "THAT'S A
FAIR SCORE. STUDY HARDER!"
540 IF S<70 THEN ? "YOU COULD DO BETTE
R THAN THAT!"
550 ? :? "GOODBY! ";NAME$
560 END
570 FOR Q=12 TO 23
580 POSITION 0,Q
590 ? BL$(1,39);
600 NEXT Q
610 RETURN
1000 DATA 4+2,6
1010 DATA 1+1,2
1020 DATA 3+4,7
```

TYPO TABLE

Variable checksum = 244641

Line num range	Code	Length
10 - 120	FX	511
130 - 240	IS	288
250 - 360	UU	449
370 - 480	IJ	369
490 - 600	HR	392
610 - 1020	FE	39

Correction.



It's simple.

New **Spell Perfect**™ from LJK does what your second grade teacher wanted to do. It corrects your spelling. Used with Letter Perfect,™ it checks your work against an expandable dictionary. Gives "sounds like" suggestions. Counts words. Add words. Technical terms. States and towns.

You can do it all, all-in-one dictionary.

When it comes to practical software for Ataris, Apples and look alikes, Spell Perfect is simple to learn. And hard to beat. Ask your dealer for a demonstration, or write LJK for more information.

LJK™

LJK, INC. • 7852 Big Bend Blvd., St. Louis, MO 63119 • (314) 962-1855

ATARI TAPES

Leg up on languages

by PAUL WAXMAN

¿Habla español? No? Neither do I, but I've always envied my bilingual friends' ability to converse in more than one language. So it was more than idle curiosity that prompted me to take a close look at Atari's **Conversational Spanish** program. Perhaps Atari and I could do in several weeks what my high school teachers couldn't do in four years – make me bilingual.

The Conversational Spanish program is one of four programs in Atari's Conversational Language series. The others are Conversational French, Italian, and German. Each set comes with five cassette tapes and a coursebook, and sells for $59.95. The series was created and developed by EMI Audio Visual Services Ltd. and the Longman Group Publishing Co. Unlike Atari's previous venture into computerized education, the "Talk and Teach" series, the Conversational language programs require only the Atari BASIC cartridge.

The course is designed as a first step for anyone fourteen years old or older who wants to begin to study Spanish. By the time you complete the course's ten units, you will be ready to learn to speak, read, write, and understand Spanish.

The course is divided into ten units: About you; Yourself and others; Please; Finding the way; Where; When; Describe it; I like; I want; and Actions. Each of the cassettes contains two units (one on each side). The coursebook is divided into the same ten units and also contains a two-page summary of grammatical rules, a list of selected vocabulary words, and a suggested study routine. Two or three "learning phases" make up each unit.

Although the tapes can be used independently from the coursebook, the book is designed to be used as each section of the program is completed. Each unit in the book is divided into three sections. The Study section reviews the vocabulary taught in that unit and introduces appropriate grammatical rules. The Practice section consists of written exercises that test your knowledge of the material. These exercises can include fill-ins, matching problems, the forming of sentences, and other similar work. The Activities section includes exercises in reading and writing, such as answering questions about a map, ordering food from a menu, or writing a letter to a friend.

The computer portion of the course is also presented in several different ways. Sometimes you look, listen, and repeat. At other times, you read and type in a response to a question. After you type in your answer, the computer immediately lets you know if you are correct. If you are wrong, you get another chance.

Trying to learn a foreign language from a computer, however, can be like trying to learn about music from a book. You may end up knowing a lot about the mechanics of chords, but you'll never fully appreciate the beauty of the sounds those chords can make. Using a computer to become familiar with a foreign language can lead to similar frustrations. You may become expert at spelling and at conjugating irregular verbs, but without hearing the words spoken correctly you will not be able to learn to speak the language intelligibly.

However, Atari's ability to mix a voice track with a computer program, using the 410 Recorder, alleviates some of these problems. Atari's Conversational Spanish program shows you a written sentence, lets you hear a human (not synthesized!) voice reading that sentence, and then gives you ample time to repeat it. Sometimes you are asked to listen to a conversation between two people and then to answer questions about the conversation. At other times, you are asked to match parts of fragmented sentences shown on the screen.

Because I know almost no Spanish, I was able to look at this program from the viewpoints of both an educator and a student. I found that the course's multiple approach to each lesson virtually ensures that you will learn something about the Spanish language. However, I have several major complaints about the program.

First of all, the speakers on the tape tended to speak too quickly for a beginner to follow. According to the Spanish-speaking teachers who listened to parts of the tapes with me, their pronunciation was very good. However, I couldn't always detect subtle nuances, such as the "g" in "gusto."

Secondly, there is the question of cor-

rect as opposed to common usage. The coursebook states that either "Usted es" or "Es usted" can be used to mean "Are you?" as in "Are you Carmen Lopez?" However, "Es usted" is the preferred form.

I also found the pacing of the program to be painfully slow at times. For example, I spent almost thirty minutes learning to say "Are you Carmen Lopez from Mexico City?"

But the biggest obstacle to using the program effectively was the review feature. The program periodically asks you whether you want to continue with the next lesson or review the lesson you have just finished. But because the program cannot rewind the tape, the review must be done without the voice track, unless you are willing to start again at the beginning of the tape.

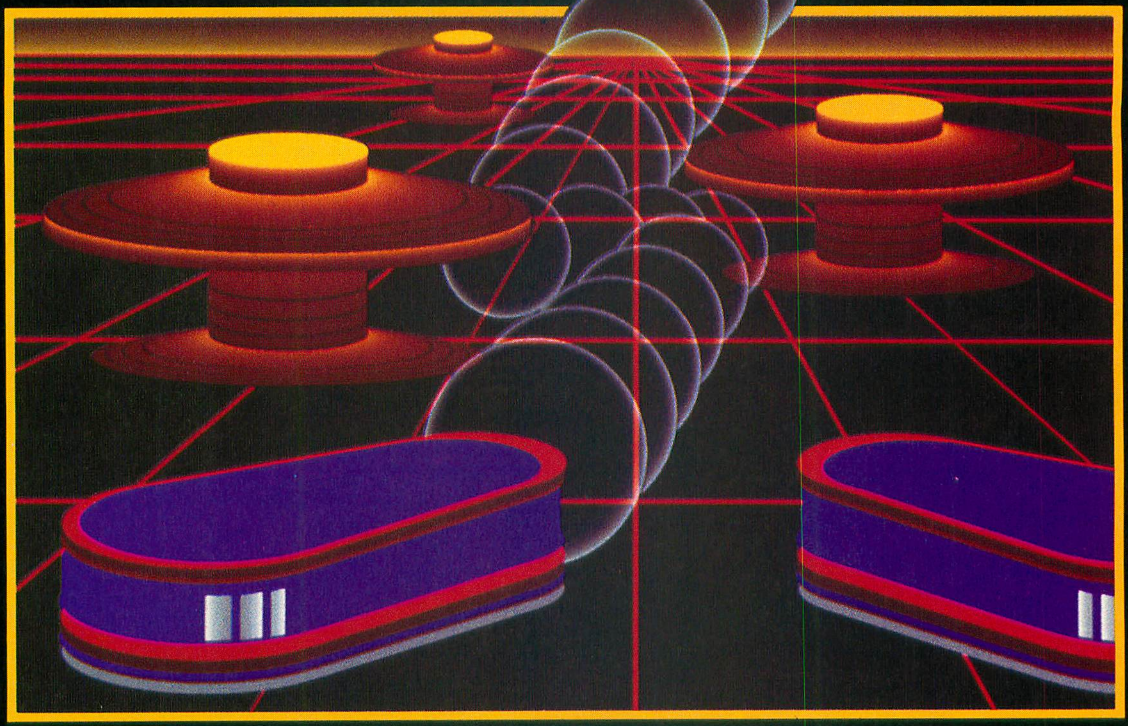
In addition, the synchronization of the cassettes and the computer makes it very difficult to stop the program and spend extra time on a sentence or concept that you find particularly troublesome. And a poorly-timed pressing of the [BREAK] key can throw the synchronization completely off, and make it necessary to start from the beginning of the tape.

Finally, I would like to have seen more extensive coverage of grammatical structures within each unit of the coursebook. Instead, I had to turn to the back of the book constantly to find the specific rules of grammar that applied to the units I was studying.

As stated in the introduction to the coursebook, the goal of this program is to motivate you to learn the language. You will certainly learn enough Spanish to fumble your way through most everyday experiences, which isn't such a bad idea. After all, most people who visit the United States make an effort to gain at least a rudimentary knowledge of English. Conversational Spanish did not make me truly bilingual, but it did give me an introduction to and an appreciation for another language. That's a good first step.

For Atari computers with 32K or more memory

TAKE A BREAK!



WITH NIGHT MISSION

PINBALL

You deserve the best. You've earned it. Now reward yourself with a session of **Night Mission PINBALL**, the most realistic and challenging arcade simulation ever conceived! ■ Stunning graphics and dazzling sound effects put **Night Mission PINBALL** in a class by itself. Game features: multi-ball and multi-player capabilities, ten different professionally designed levels of play, and an editor that lets you create *your own* custom modes. ■ So take a break with **Night Mission PINBALL** from SubLOGIC. Winner of *Electronic Games* magazine's 1983 Arcade Award for Best Computer Audio/Visual Effects.



See your dealer . . .

or write or call for more information. For direct orders please add $1.50 for shipping and specify UPS or first class mail delivery. Illinois residents add 5% sales tax. American Express, Diner's Club, MasterCard, and Visa accepted.

Order Line: 800/637-4983

subLOGIC
Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995

POCKET CALENDARS FOR 1984

Use your printer to keep appointments

by BILL LUKEROTH



SYSTEM REQUIREMENT: 16K, PRINTER

I was watching my 820 printer lazily churn out a program listing one day, when it occurred to me that, other than listing programs and printing address labels, I hadn't found much use for it. This state of affairs offended my frugal Teutonic sensibilities. After a little reflection, I realized that the four-inch-wide strips of paper that the Atari 820, 822, 1020, and Alphacom printers use are about the right size for a pocket appointment calendar. I then created a perpetual calendar program, which in a burst of non-creativity I titled "Calendar."

Calendar asks you for a month (use the month's number, not its name) and a year. Once you answer, the printer will start up almost immediately, so make sure that you turn it on *before* you answer the prompts. If all goes well, the result will be a 9″× 4″ calendar, with the dates correctly aligned with the days of the week (for example, January 1, 1984, will be a Sunday). Write your appointments in the spaces provided, then fold up the strip and put it in your wallet or

purse.

As printers go, the 820 is pretty pedestrian, but it does have one rather remarkable feature: it can print sideways. This is another thing I hadn't previously found much use for, but I cured that situation by writing a second calendar program, "Sidewinder." Sidewinder isn't nearly as intelligent as Calendar, but the calendar it produces is prettier and has more of a high-tech look. I think the 820's "sideways" characters are better looking than its normal ones.

Although it was designed for 40-column printers, Calendar should also run on any 80-column machine; however, because of the sideways characters, Sidewinder will only run on an 820. In either case, if you have a printer you now have one less excuse for forgetting an appointment in 1984!

Listing 1

```
95 REM *****CALENDER*****
96 REM *BY BILL LUKEROTH*
97 REM * ANTIC MAGAZINE *
98 REM * JANUARY, 1984   *
99 REM ******************
100 DIM B$(36),T1B$(36),DL$(36),DAY(42
),M$(10),P$(2)
110 B$="|    |    |    |    |    |
|    |"
120 DL$="------------------------------
-------"
130 BASEYEAR=1950:STARTOFF=0:LEAPFLAG=
0:CUROFF=450
140 GRAPHICS 0:GOTO 5000
440 REM ++++++++++ curoff ++++++++++
450 POKE 752,1:RETURN :REM turn cursor
 off
800 READ M$,ML
810 IF ML=3 AND YEAR/4=INT(YEAR/4) THE
N ML=2
830 OFFSET=TEMPOFF
840 IF L2<MONTH THEN TEMPOFF=(OFFSET+3
)-ML
845 IF TEMPOFF>=7 THEN TEMPOFF=TEMPOFF
-7
850 RETURN
5000 ? "CALENDAR FOR WHAT MONTH(INPUT
NUMBER, NOT NAME)";:INPUT MONTH
5010 ? :? "FOR WHAT YEAR";:INPUT YEAR:
LEAPFLAG=0
5020 IF YEAR<BASEYEAR THEN ? :GOTO 501
0
5030 DIF=YEAR-BASEYEAR:TEMPOFF=STARTOF
F:TEMPYEAR=BASEYEAR
5040 FOR L=1 TO DIF
5050 TEMPOFF=TEMPOFF+1:TEMPYEAR=TEMPYE
AR+1
5060 IF TEMPYEAR/4=INT(TEMPYEAR/4) THE
N TEMPOFF=TEMPOFF+1
5070 IF TEMPOFF>=7 THEN TEMPOFF=TEMPOF
F-7
5080 NEXT L
5090 IF YEAR/4=INT(YEAR/4) THEN LEAPFL
AG=1
5100 IF LEAPFLAG=1 THEN TEMPOFF=TEMPOF
F-1:IF TEMPOFF<0 THEN TEMPOFF=TEMPOFF+
7
5110 FOR L2=1 TO MONTH
5120 GOSUB 800
5150 NEXT L2
5160 OFFSET=TEMPOFF
5165 T1B$=B$
5166 ? "DO YOU WANT THIS ON THE:":? "
 SCREEN":? "    PRINTER":? :? "(TYPE S
 OR P)":INPUT P$
5167 IF P$<>"S" AND P$<>"P" THEN ? :GO
TO 5166
5168 P$(2,2)=":":OPEN #1,8,0,P$:GOSUB
CUROFF
5170 ? #1;"                   ";M$:? #1;
5180 ? #1;"    S    M    T    W    T
 F    S"
5190 ? #1;DL$
5200 FOR L3=1 TO 42:DAY(L3)=0:NEXT L3
5210 FOR L4=1 TO 42
5220 DAY(L4)=L4-OFFSET:IF DAY(L4)<0 TH
EN DAY(L4)=0
5230 IF L4-OFFSET>30-ML THEN L4=1:POP
:GOTO 5250
5240 NEXT L4
5250 LD=-2
5255 FOR L5=1 TO 42
5260 LD=LD+5:IF LD>33 THEN ? #1;T1B$:?
 #1;B$:? #1;B$:? #1;B$:? #1;B$:? #1;B$
:? #1;B$:? #1;DL$:T1B$=B$:LD=3
5270 IF DAY(L5)=0 THEN 5400
5280 IF DAY(L5)<10 THEN T1B$(LD+1,LD+1
)=STR$(DAY(L5)):GOTO 5400
5290 T1B$(LD,LD+1)=STR$(DAY(L5))
5400 NEXT L5
5410 ? #1;T1B$:? #1;B$:? #1;B$:? #1;B$
:? #1;B$:? #1;B$:? #1;B$:? #1;DL$
9999 END
10010 DATA JANUARY,0,FEBRUARY,3,MARCH,
0,APRIL,1,MAY,0,JUNE,1,JULY,0,AUGUST,0
,SEPTEMBER,1,OCTOBER,0,NOVEMBER,1
10020 DATA DECEMBER,0
```

TYPO TABLE

Variable checksum = 825943

Line num range			Code	Length
95	-	450	UH	433
800	-	5050	FY	324
5060	-	5167	AC	344
5168	-	5260	MG	524
5270	-	10020	IR	355

Listing 2

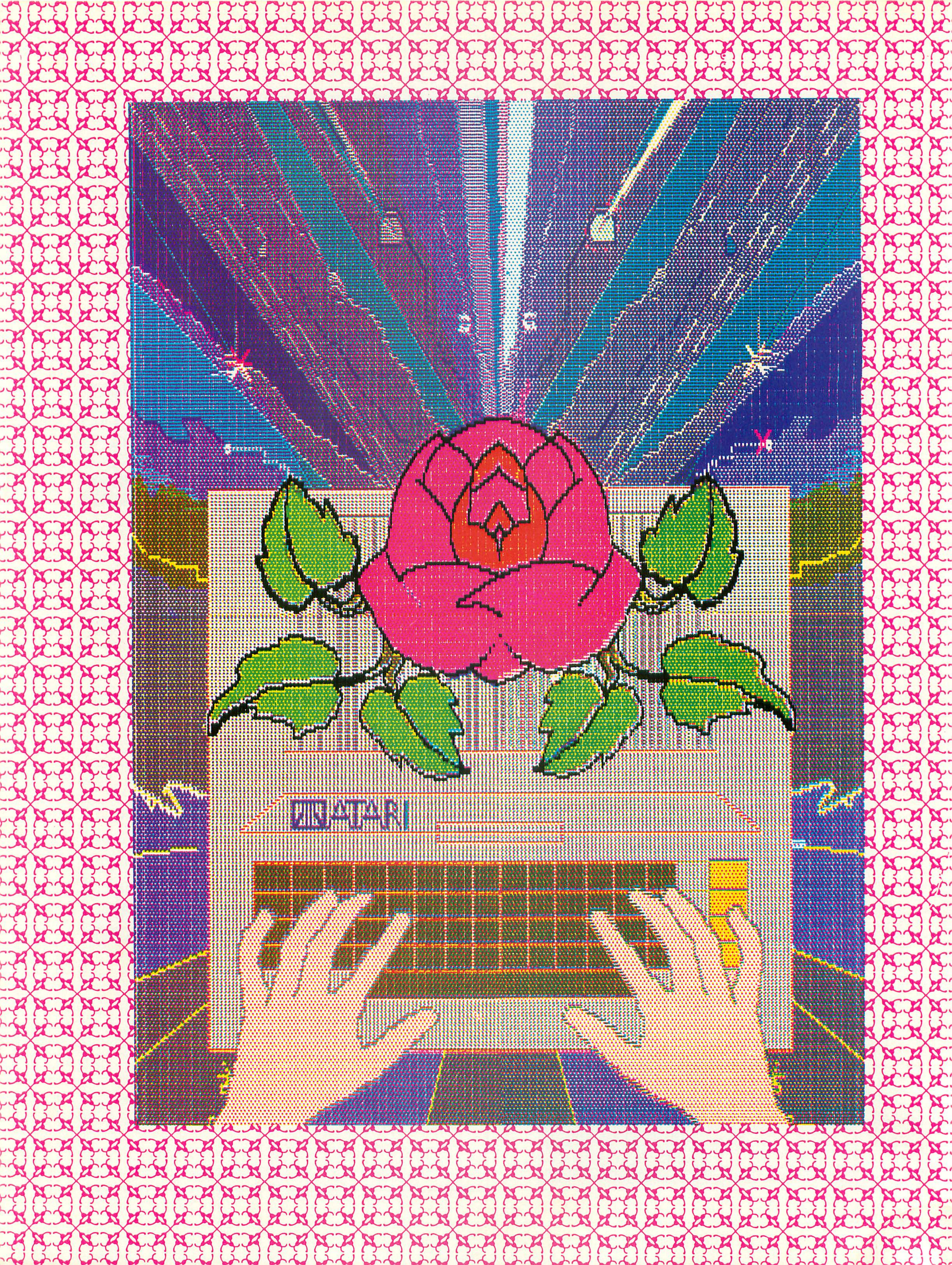
```
195 REM ****SIDEWRITER****
196 REM *BY BILL LUKEROTH*
197 REM * ANTIC MAGAZINE *
198 REM * JANUARY, 1984   *
199 REM ******************
200 DIM M$(10),DY$(7),NAME$(30):DY$="S
MTWTFS":C=0
300 GRAPHICS 0:? :? "            MONTHLY
 CALENDER"
```

```
31Ø OPEN #1,8,Ø,"P:":OPEN #2,8,83,"P:"
315 ? "WHAT'S YOUR NAME";:INPUT NAME$
:IF LEN(NAME$)>1Ø THEN NAME$=NAME$(1,1
Ø)
317 NAME$(LEN(NAME$)+1)="'S CALENDAR F
OR "
32Ø ? :? "WHAT MONTH(ENTER NUMBER,NOT
NAME)";:INPUT M
33Ø ? :? "WHAT YEAR";:INPUT Y:IF Y>99
THEN Y=Y-19ØØ
335 ? :? "HOW MANY DAYS IN THIS MONTH"
;:INPUT LAST
34Ø ? :? "THE FIRST FALLS ON:":? "1.SU
NDAY":? "2.MONDAY":? "3.TUESDAY":? "4.
WEDNESDAY":? "5.THURSDAY"
35Ø ? "6.FRIDAY":? "7.SATURDAY":? :? "
(ENTER 1,2,3,ETC.)":INPUT N
36Ø IF N<1 OR N>7 THEN GRAPHICS Ø:? :?
 "YOUR CHOICE MUST BE BETWEEN 1 AND 7"
:? :GOTO 34Ø
38Ø IF N=1 THEN LP=1:GOTO 4ØØ
39Ø LP=N-((N-1)*2)
4ØØ FOR L=1 TO M:READ M$:NEXT L
41Ø ? #1;"     ";NAME$;M$;" ";Y:GOSUB 5Ø
ØØ
415 FOR DAY=LP TO LP+6
42Ø ? #2;"=Ø";:IF DAY<1 THEN ? #2;"Ø";
:GOTO 44Ø
43Ø ? #2;DAY;
44Ø ? #2;" =";:IF DAY+7<1Ø THEN ? #2;"
Ø";
45Ø ? #2;DAY+7;" =";:IF DAY+14<1Ø THEN
 ? #2;"Ø";
46Ø ? #2;DAY+14;" =";:IF DAY+21>LAST T
HEN ? #2;"ØØ";:GOTO 473
47Ø ? #2;DAY+21;
473 ? #2;" =";:IF DAY+28>LAST THEN ? #
2;"ØØ";:GOTO 48Ø
475 ? #2;DAY+28;
48Ø ? #2;" =";:IF DAY+35>LAST THEN ? #
2;"ØØ";:GOTO 485
483 ? #2;DAY+35;
485 ? #2;" = ";:C=C+1:? #2;DY$(C,C)
49Ø FOR L2=1 TO 6:? #2;"=   =   =   =
   =   =   =":NEXT L2
5ØØ GOSUB 5ØØØ
51Ø NEXT DAY
999 END
5ØØØ FOR LØ=1 TO 6:? #1;" ";:NEXT LØ:F
OR L=1 TO 34:? #1;"-";:NEXT L:? :RETUR
N
1ØØ1Ø DATA JANUARY,FEBRUARY,MARCH,APRI
L,MAY,JUNE,JULY,AUGUST,SEPTEMBER,OCTOB
ER,NOVEMBER,DECEMBER
```

TYPO TABLE

Variable checksum = 197977

Line num range	Code	Length
195 - 33Ø	WM	5Ø5
335 - 42Ø	KG	5Ø6
43Ø - 49Ø	SP	526
5ØØ - 1ØØ1Ø	EF	2Ø6


ATARI

THE BLOSSOMING OF COMPUTER ART

Create it with your Atari and a printer

by CHARLES BENNETT

This article is probably most meaningful for professional artists who are interested in using the Atari computers to create works of computer art. But it may also inspire young Atari artists to think along these lines. As a commercial artist, I've had an avid interest in computers for almost three years now, but I never really paid attention to the artistic potential of the home computer until ANTIC's Cover Art Contest came along (ANTIC, August 1983). At that time, Jo Ann Brissenden, a close friend, convinced me to team up with her and enter the contest. The result was "The Blossoming of Computer Art."

Our objective was to create a work of computer art that would retain its integrity as computer art. That is why you can still see the dots in the finished piece. However, we also wanted to clearly convey to the viewer more detail than the simple "needle-point" look can provide.

One possible route would have been to either photograph a full-screen picture on the monitor, or produce a printout version of the picture that was created on the screen.

Unfortunately, either one of these methods usually results in a picture that is too crude, even when reduced, to meet our goal of clear visual communication. This is true with most, if not all, affordable microcomputers. We were able, however, to quadruple the amount of detail that is usually provided by such computers by using the Atari 800 along with Micro-Painter, Color Print and Graphic Master programs. A Gemini 10 printer was also used.

You'll notice that our drawing is very symmetrical. The top-right-hand quadrant is almost identical to the top left, and the lower-right-hand quadrant is very similar to the bottom left. In fact, we drew only two of the four segments to begin with. Final embellishments, such as the Atari logo, were added only after Graphic Master was used to mirror-image (or flop) the two original drawings and then to save the four resulting drawings. These embellishments gave the left and right hand sides of the final picture their individual differences.

Draw a line through the center of our picture horizontally, and then vertically, and you'll see the four Atari full-screen pictures that were used to create the final effect. Note that each rectangular screen is vertical in the final picture. We had to turn our monitor on its side to visualize how each screen-full would look as part of the completed drawing. (This can be confusing, though, because it changes the cursor-movement sequence. "Left" becomes "Up," and so on.)

Our first step was to come up with a rough penciled sketch of the entire picture (see Figure 1). This was then enlarged, inked on acetate, and cut into four sections. Next, we taped the bottom-right quarter of the clear acetate drawing to our monitor's screen for quick layout reference (you can also draw directly on the glass with a washable marker).

Using Micro-Painter, we then drew this portion of the drawing electronically and saved it. At this stage, we also made a printout so we could ensure that the connecting lines from one full screen would align with those from the other (original) screen. Once this was accomplished, we created a mirror image of the screen with Graphic Master, saved it and then ran a printout.

The opposite (top-right) section of the acetate drawing was handled in the same way. This gave us (after mirror-imaging or flopping the two original pictures) four separate full-screen pictures to print out and splice together. It's important to keep these drawings in outline or skeletal form, because it is difficult to get them to line up properly from top to bottom (see Figure 2). We found that we were usually a few cursor-dots off vertically, and this is something that needs to be checked for and corrected before you fill in the outlines with color.

Next, it was time to figure out a color scheme. We initially did this by marking up a thumbnail sketch on paper with color makers, because changing colors midway through a drawing can be much more time consuming than simply planning ahead. We decided, for example, that a red outline was needed to convey

continued on next page

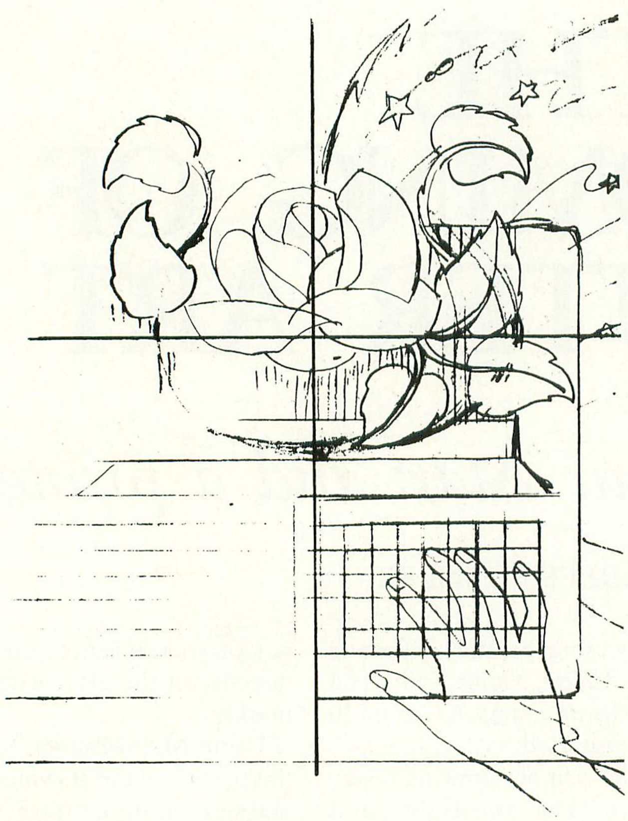


Figure 1. Preliminary sketch.

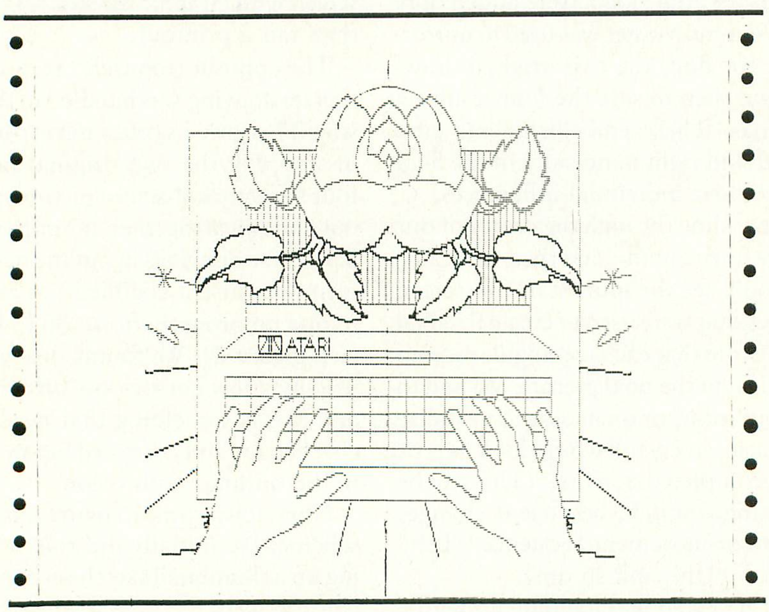


Figure 2. Skeletal layout to check registration.

a sense of excitement and to identify the keys on the computer keyboard. The machine's light-mocha color also has red dots in it. If you try to draw a red line through such a color-filled area, though, you'll find that your line will stop at every fifth pixel. It's this kind of frustration that can take a lot of the fun out of computer art, so plan ahead!

Next, we checked our work with the Color Print program, which gave us a colored carbon copy of our work in progress.

Once we had both (original) screens color-filled and mirror-imaged, and had printed out copies of the four resulting screens, we taped them together and checked for registration. Again, using the Color Print program and its colored carbons, each screen was then color separated and printed out. Because four color separations were needed for each of the four segments of the picture, this meant that sixteen different screens (and printouts) were needed. This was the final testing phase before we got a "color key" and film negatives ready for the printer (more on this later).

Now, look even more closely at the original picture. You'll see the tiny dots that tell you it was computer-generated. They are much more detailed and much less blurry than in many examples of color computer art, because this picture was not photographed off a screen or reproduced in color using a dot-matrix printer and the color carbons supplied in the Color Print package. The colors in the finished Color Print version of the picture (used for testing purposes only) are darker than those in the original screen versions, because the color carbons do not provide accurate color representation. Also, the sheet used in the Color Print test had to be run through the printer four separate times (once for each primary color). You have to be very careful to get exact registration on each run, and each one is a calculated risk — frankly, you'd have better luck in Las Vegas! As a result, the carbon process won't do for reprinting your drawing, but it does give you a good, inexpensive preview of your finished work.

To create art with your Atari that can be professionally reproduced, you may need to learn something about the process that is used to produce full-color pictures for books, magazines and con-

sumer packaging. It's a very different process from your color monitor's use of red, blue, and green to produce pictures on the screen. Professionally color-separated pictures (and photographs) are made up of four separate colors: red, yellow, bue and black. Each color is contained on a separate transparency; a set of four transparencies is combined to create a finished four-color picture. (The Color Print program, by the way, handles this color-separation phase for you.)

At this point, we spliced our 16 printouts together into four complete black-and-white pictures. We taped the pictures together very carefully, making sure to match up the dots that overlapped. We did this by taking the shade off a lamp, holding two printout sheets together over the naked light bulb, matching the overlapping dots, and then taping the two sheets together. Each sheet was then marked with one of the four appropriate colors: red (magenta), yellow, blue (cyan) and black. The result of all of this work was four complete black-and-white representations of the picture, one for each color (see Figure 3). These were the basis for the actual color separations. (Pre-separated artwork such as this can save between $75.00 and $300.00 — depending on the color separator — at the printing stage. This can make your artwork even more competitive commercially.)

The next step was to get a "color key" of our drawings. At this stage of the process, a professional printer took the four pictures we'd created, made film negatives and then produced a color key. This involved the printing of four colors on four thin-acetate sheets. These were then carefully overlaid to attain exact registration. This gave us a chance to preview our work before the actual printing process began. The process is similar to that involving Color Print's carbon overlays, except that it is much more accurate.

This effort begins to seem much more worthwhile when you consider that our finished picture, created on a dot-matrix printer that retails for about $400.00, produced a commercially-printed color picture with better dot fidelity than that provided by Seiko's new 150-dot-per-inch thermal-transfer printer ($8950.00) or a standard ink-jet printer (approximately $600.00). No matter how fine a four-color computer printer is, artwork produced on it must be rescreened by the color separator to be made ready for commercial printing. This Atari artwork, however, did not need to be rescreened.

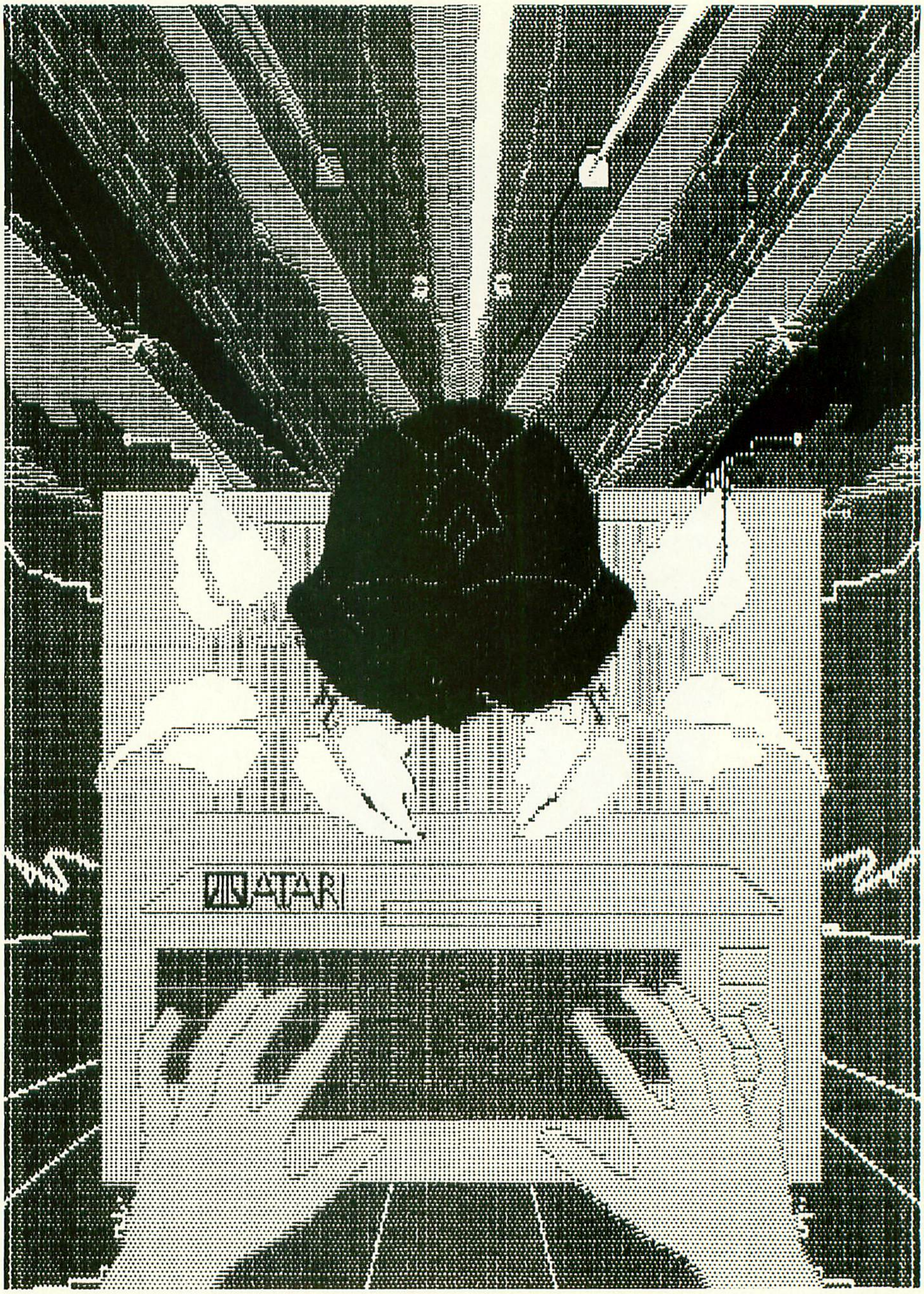


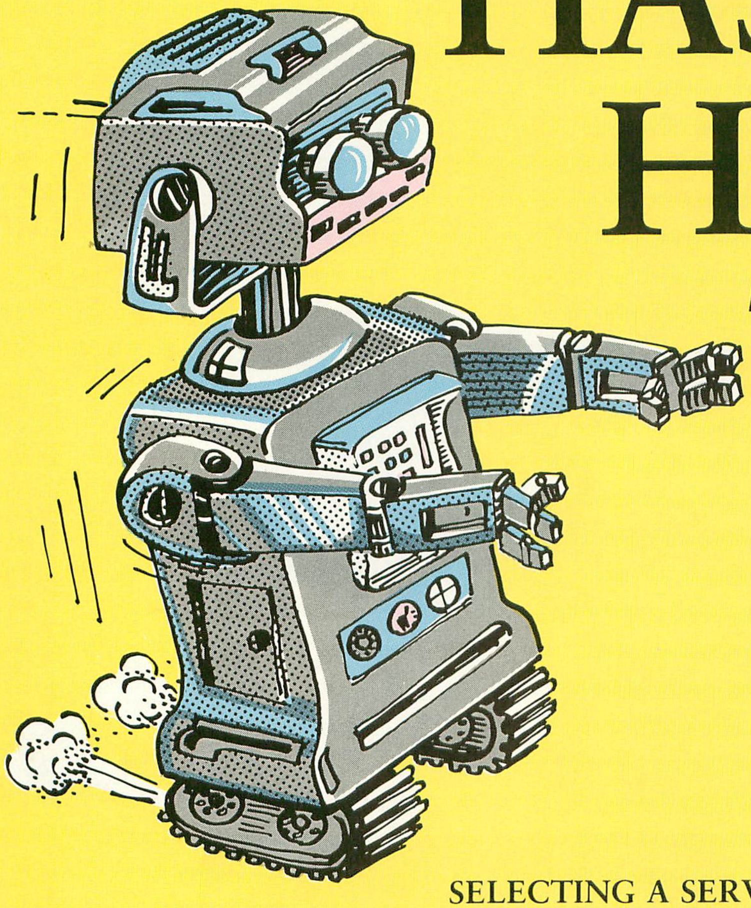
Figure 3. Sample black-and-white printout (for the color "magenta").

Finally, I have several suggestions for those of you who are ready to try this new technique. First, make sure that you use a new printer ribbon when you print the final four pictures.

Second, explain the technique to the professional printers and color separators you deal with. Otherwise, they won't have the foggiest idea of what you're trying to do. Remember that this is a brand new category of art that very few people are even aware of; show them this article if they don't understand the technique.

One final note: Artistic accidents happen all the time, and often help the creative process. Be ready for them! Keep a lot of diskettes handy, and always remember to back up your work. Backup diskettes are particularly helpful when you decide to change colors in a color-filled area — it's usually easier to simply go back to a previously-saved disk file.

HAS YOUR HUGGED TODAY?



by EVAN ROSEN

Last month, in "Anatomy of a Robot," we talked about creating motion with your computer. This time, we'll actually do it. We'll need:

- a servo
- a joystick-port connector
- alligator clips or a soldering iron and solder
- a lantern battery and two diodes, or a 5-volt power supply
- an Atari 400/800 computer with BASIC or Forth

Because home computer enthusiasts are not necessarily electronics hackers as well, I'll provide you with as much detail as possible. By following these instructions, you should be able to obtain a simple assortment of parts and to learn an easy way to make them move.

Evan Rosen is the co-author of Val-FORTH from Valpar International.

SELECTING A SERVO

A servo is a small, commercially-available plastic box that contains a DC motor, a gear train, and an electronic assembly. Its largest dimension is usually one to three inches, and it has at least one short protrusion, called an output shaft, which rotates when the servo receives the proper signal.

Hobby servos traditionally have been used in radio-controlled (RC) models. The ailerons and landing gear on RC airplanes, the rudders on RC boats, and the steering mechanisms on RC cars are typical applications. Because these models are so popular, it's likely that a nearby hobby shop can sell you a servo. In addition, some of the RC-model magazines can provide you with the names of mail-order houses from which you can order servos and related parts.

When selecting your first servo for robotics, buy the least expensive one you can. You should be able to find a medium-sized assembled servo for around $18, and possibly for a few dollars less if you've got a discounter in town or you order by mail. Unless you're an avid tinkerer, you should stay away from build-it-yourself models.

Servos are rarely able to rotate in a complete circle. Instead they typically have a limited "rotary output," which ranges from 90 to 270 degrees. The 180-degree models are the most common and are fine for our purposes.

If you have a choice of connectors on your servo, get the Molex variety. They're less expensive, and you'll probably just cut them off anyway and use the wire ends.

Don't buy "retract" servos — they're only two-position devices. And don't worry about too much torque (turning strength) ratings on your first servo. The garden-variety type has something like 20 ounce-inches of torque, which is about right for operating a gripper. There'll be plenty of time later to spend $50 on a monster with ball-bearing output and enough torque to reel in a bowling ball on a half-inch shaft.

The servos that were tested (at random) with the program in this issue were World, Ram, Kraft, and Royal. There are at least ten other brands available, and any one of them should work as well. Check to see that there are three wires extending from the servo you buy. These are usually red (5-volt input), black (ground), and a third color (pulse input).

ROBOT YOU

For the time being, don't get one of the larger five-wire models.

Also, don't rotate the servo's output shaft by hand if you can avoid it. This can create slack in the gear train and result in jittery performance.

One final warning: Servos have motors that contain permanent magnets and that generate magnetic fields! Keep them at least three feet from your disks.

JOYSTICK-PORT CONNECTORS

Fortunately, Atari used a standard connector on its joystick ports. We need a DE-9S female, which you have find at many electronics stores. And if you have a surplus electronics store nearby, you may be able to pick up a DE-9S for as little as $.50 to $1.

In fact, you may even be able to find a plain Atari joystick cable — one that has a molded connector at one end and six wires protruding from the other end of a three-foot cable. There seem to be a lot of these around. On such a connector, the wires should be lined up as follows: White-pin 1, blue-2, green-3, brown-4, orange-6, and black-8. This lineup shows that the cable was intended for a joystick rather than a paddle. If available, these surplus connectors may cost only a dollar each. Ask the dealer to let you use an ohmmeter to check continuity, or to allow you to swap if you get a bad cable. It never hurts to mention that you're making a robot.

GETTING 5 VOLTS

As you shop for a servo, you'll notice that they usually specify 4.8 volts. This is because among hobbyists servos are almost always run from four nickel-cadmium or "nicad" batteries in series. Each of these has a voltage of about 1.2 volts under load. A five-volt servo is fine, too, and you can even go up to 5.2 volts. You shouldn't go much higher, though, since we'll be reading a 5-volt logic level from the computer.

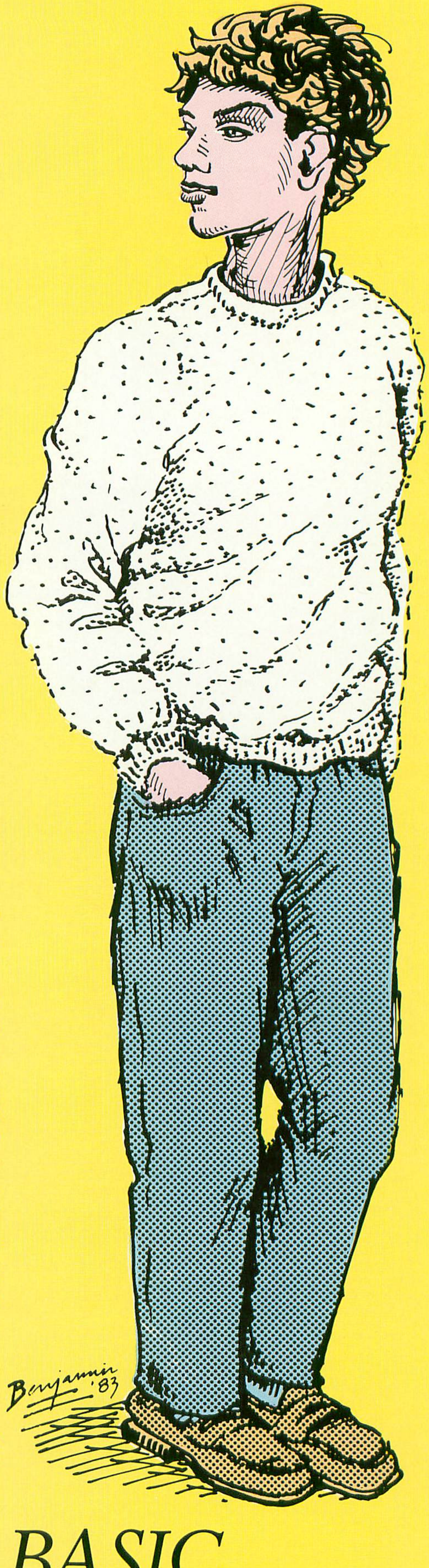
If you already have a regulated 5-volt source with a rating of 300 milliamps or more, you need go no further; if you have four nicads and a charger, you're also in good shape. If not, how about a lantern battery — that reliable, square-bodied, 6-volt dry-cell battery? This power source can be purchased just about anywhere for about $3.00. Even if it isn't one of the more expensive alkaline types, a battery like this will probably last you a week, which is more than enough time to find out if you've caught robot fever.

Since this battery is nominally 6 volts strong, and is actually closer to 6.2 volts in strength when it's fresh and under a light load, we have to "throw away" about a volt. We can do this by putting two diodes in series and connecting the free cathode of the pair (the end without a marking stripe) to the + 6-volt terminal on the battery. The free anode of the pair now becomes the + 5 supply for our servo. This isn't very precise, but it will work. Use 1N4001 diodes, since they're inexpensive and readily available.

HOOKING UP

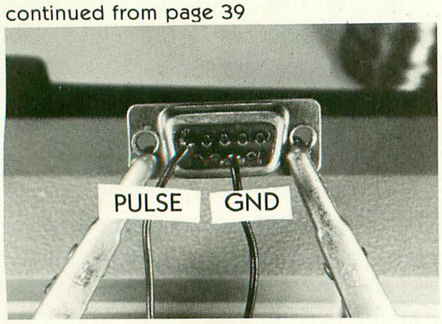
The following four connections, each highlighted by "(X#X)" in the text, should be made with the joystick-port

continued on page 41



Creating motion in BASIC.

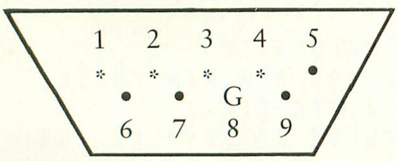
continued from page 39



Close-up of connector with wire.

connector removed from the computer. Be careful when making these connections. If you misconnect the power-supply leads and then plug into your Atari when it is turned off, you can damage the PIA chip. In general, it is best if you plug powered, home-made connectors into your computer only while it is on. The same holds true for unplugging such connectors.

Let's review the pinout on the joystick-port connectors to be sure that we're in step. Facing the port, or looking at the back of a D-9 connector, the pins are numbered like this:



(The pins marked with an "*" can be used for output. We'll be using pin 1 on Port 2.) The proper connections are as follows:

- Ground: The power supply, the servo, and your computer will share a common ground. To accomplish this, we connect the negative terminal of the power supply to the black wire of the servo (X1X), and to pin 8 of the connector (X2X). If you're using a surplus joystick cable, pin 8 *should* be the black wire; however, it would be best to check this by using an ohmmeter. Remember that the pins facing the front of the molded connector are the mirror image of the above diagram.
- Servo pulses input: Connect the third wire of the servo (the wire that is neither black nor red) to pin 1 of the connector (X3X). If you're using a molded plug, pin 1 should be the white wire, but (again) double-check with your ohmmeter or another continuity checker to be certain.
- Servo 5-volt connection: Finally, connect the positive end of the 5-volt supply to the red wire of the servo (X4X).

When the wiring has been completed and double-checked, put the assembly aside. We'll type the program in next, and then plug the connector into the joystick port and see what we've got.

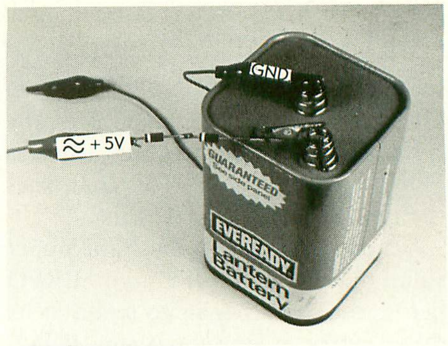
DRIVER PROGRAM

As was mentioned in the last issue, we want to set up the driver program as an interrupt-driven, machine-language background task. This will allow BASIC or Forth to run normally as a foreground task, albeit slightly slower than usual.

We want to send the servo a pulse periodically (60 times per second) from pin 1 of Port 2. This pulse should have a width in the range of about one to two msec. We'd like to do this with a relocatable machine-language routine, which we'll call the servo-driver routine. Finally, we'd like to have a simple way to send new orders to the servo-driver routine from our high-level language. Such a BASIC listing is provided at the end of this article. A Forth version, along with some sample programs for both versions, will appear in the next issue.

BASIC LISTING NOTES

Enter the BASIC program and save it to disk or cassette if you have one of these devices. Then plug the joystick-port connector on your servo assembly into Port 2, and RUN. Your servo should now be oscillating between its extremes every



Battery with clips

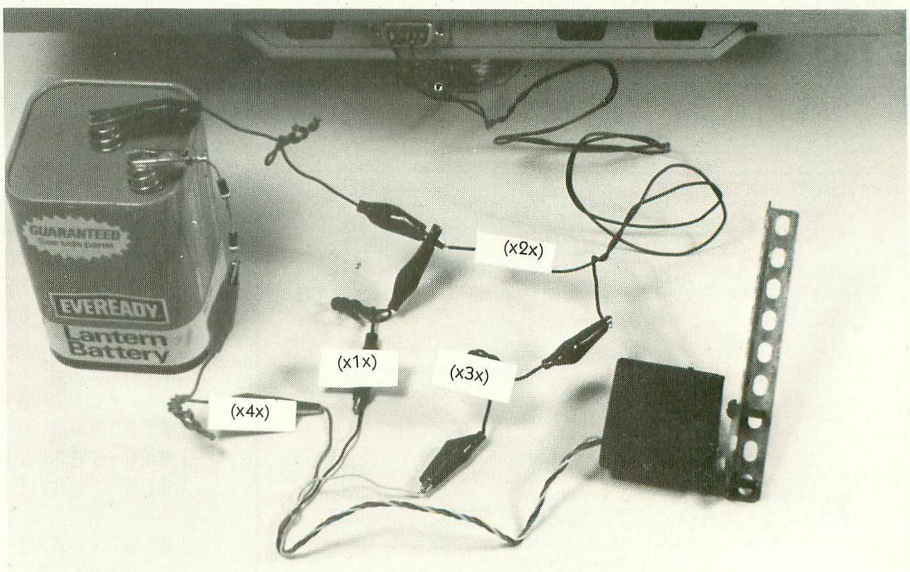
four seconds or so. If it isn't oscillating at all, check your wiring again. But if you're getting motion, we can now "tune" the program.

To adjust the program to your particular servo, you must modify two values, the starting point and the width of travel. Press [BREAK] to exit from the oscillation loop and then do a POKE SERVO, 0. The servo will probably move once when you do this. We want it to move to its "0" position, which is as far as it can go clockwise (from your vantage point, facing the output shaft). However, if the program directs the servo to go past its "0" position, it will be under a continuous strain, and will vibrate. Touch the servo's case. If you feel any vibration, type in:

POKE OPULSE,PEEK(OPULSE) + 2

and check the servo's case again. If it is still vibrating repeat the above line until the vibration stops. Then repeat the line twice to give your servo a little breathing room and then type:

continued on next page



Hookup

PRINT PEEK(OPULSE)

to get the proper value at location OPULSE. Next, edit line 170 of the BASIC program so that this value is POKEd into OPULSE. This completes the first part of the calibration.

We next want to establish the upper value that we can POKE into SERVO before the servo begins to strain at its other extreme. Start by typing POKE SERVO,150. This should move the servo most of the way around. Now type:

POKE SERVO,PEEK(SERVO) + 5

and feel the case to see if the servo has begun to strain yet. Repeat the above line until the case begins to vibrate; then back off a bit by typing:

POKE SERVO,PEEK(SERVO)–2

and repeat this line until the strain is relieved. Then repeat the line twice to provide a margin, as we did a moment ago when setting OPULSE. Then type:

PRINT PEEK(SERVO)

and jot this value down. Edit line 1010 so that this value is assigned to TOP for the oscillation demo. Then type GOTO 1000 to resume the demo, which should now be moving the servo between its extreme points. The range of values that can be POKEd into location SERVO is from 0 to TOP. Other values will either cause the servo to strain or will generate an error condition (if they are outside the range of 0 to 255). Calibration of your servo is now complete. Save your listing.

When you first ran the program, by the way, you actually installed a short machine-language routine. This generated the pulses that the servo needed to see, based on the value POKEd into SERVO. The vertical blank interrupt vector was then pointed to this routine so the appropriate pulses would be generated 60 times per second. This pulse routine is relocatable, and is installed starting at the value assigned to SERVO in line 140. Currently, it is at 1536 decimal, which is the beginning of Page Six of memory (unused by BASIC). If you'd like to store the routine somewhere else, simply change the value in line 140 to the installation address of your choice. Pick an address that is divisible by 256 so the routine won't cross a page boundary — this could alter the timing of several branch instructions, which could, in turn, throw off calibration.

Once the pulse routine has been installed, the whole installation program can be discarded by typing NEW. The only data you need for your subsequent program are the value you assigned to SERVO, which is where you will POKE your position values, and your TOP value for this servo.

Next month we will look at a Forth version of this listing, along with sample programs for BASIC and Forth, and will consider both multiple-servo versions and low-cost sensors.

```
95 REM *****SERVO******
96 REM *BY EVEN ROSEN *
97 REM *ANTIC MAGAZINE*
98 REM *JANUARY, 1984 *
99 REM ****************
100 REM PGRM JOY 2 FOR OUTPUT
105 TEMP=PEEK(54018)
110 POKE 54018,TEMP-4
120 POKE 54016,16
125 POKE 54018,TEMP
130 REM STORE SERVO DRIVER
140 SERVO=1536:OPULSE=SERVO+1
145 ENTRY=SERVO+2:PTR=ENTRY
150 REM SERVO IS PAGE 6 BUT
155 REM MAY BE RELOCATED
160 POKE SERVO,128:REM MIDPOINT
165 REM MAY BE RELOCATED
170 POKE OPULSE,100:REM MIN PULSE
175 POKE PTR,174:PTR=PTR+1
180 POKE PTR,OPULSE-INT(OPULSE/256)*25
6
185 POKE PTR+1,INT(OPULSE/256):PTR=PTR
+2
190 FOR I=1 TO 12
195 READ X:POKE PTR,X:PTR=PTR+1
200 NEXT I
205 DATA 232,169,16,141,0,211,234,234,
202,208,251,174
210 POKE PTR,SERVO-INT(SERVO/256)*256
215 POKE PTR+1,INT(SERVO/256):PTR=PTR+
2
220 FOR I=1 TO 14
225 READ X:POKE PTR,X:PTR=PTR+1
230 NEXT I
235 DATA 232,234,234,234,234,202,208,2
49,169,0,141,0,211,76
240 POKE PTR,PEEK(58467)
245 POKE PTR+1,PEEK(58468)
250 POKE 54286,0:REM DISAB VBLANK
255 POKE 548,ENTRY-INT(ENTRY/256)*256
260 POKE 549,INT(ENTRY/256)
265 POKE 54286,64:REM ENAB VBLANK
1000 REM OSCILLATING DEMO
1010 TOP=150
1020 FOR I=0 TO TOP:POKE SERVO,I:NEXT
I
1030 FOR I=TOP TO 0 STEP -1:POKE SERVO
,I:NEXT I
1040 GOTO 1020
```

RUNNING NECK AND NEC

How to use the NEC 8023 printer

by LARRY STEINER

Atari, Inc. has good success selling home computers. Why is it, then, that Atari has not enjoyed the same positive results with their printers? The answer seems to be that most of the Atari printers have not measured up in terms of either features or performance with a number of other printers on the market. As a result, it is understandable that many Atari computer owners make use of other brands of printers, such as Epson or NEC. Of course, the old bugaboo of compatibility always arises when different brands are combined into what one hopes will be an effective, easy-to-operate system. But it need not frighten us off. So let's look at the specifics: in this case, the compatibility of the NEC 8023 printer with the Atari computers.

First of all, some time and effort will be required to set up the NEC with your beloved Atari, and of course the Atari 850 Interface (or its equivalent) also will be required to use this parallel-interface printer. A suitable cable is also required (the same type that would be used with any other Atari-to-parallel-printer combination, such as one involving the Epson or the Smith-Corona TP1).

Larry Steiner is the manager of Spectrum Video, a retail video store in Mansfield, Ohio.

Inside the NEC 8023 printer are two rows of DIP switches. They allow you to change the default conditions of the printer. We are primarily concerned with Switch 2, Position 8, at this time. This position must be changed from the factory-set position to permit a line feed with each carriage return. Simply slide the switch with a small probe or pencil tip to the "red" position.

Now the printer will no longer print text or data over and over on the same line. Be sure to study the charts in the manual that relate to the DIP switches. If you follow the instructions carefully, you can set the printer up to operate in the mode that is most convenient for you. (Be sure to turn the printer off before putting your hand inside!)

The basic set up is now complete. Now we'll consider the issue of software interaction with the printer.

Don't forget that the NEC puts many valuable features at your disposal. It would be a shame to use it for program listings alone. In fact, most programs that are part of the wide world of Atari software are compatible with this printer. However, some programs, like the old Atari Word Processor, are too inflexible to use with the NEC. Newer, more flexible programs are more likely to work, but some study is usually required to get the best possible results from a given combination.

Synapse Software's Filemanager is a very popular database manager for the Atari computers, in part because it makes printer control very easy. With Filemanager 800, for example, you simply use the escape codes listed in the printer manual when the program presents you with the blank line labeled "TITLE AND PRINTER CONTROL." If you were to type in an escape character (by pressing [ESCAPE] twice) and the letter Q on that line, for example, all subsequent output would be printed in the condensed 17-character-per-inch mode. This is a handy way to pack a lot of information onto a 8½-inch-wide piece of paper.

Many standard software packages make explicit provision for the NEC printer. Datasoft's Graphics Master, for example, works beautifully with the NEC. Some packages, however, require a little more work. Atari's Home Filing Manager is one such example.

The Home Filing Manager database presents data on the screen as dark letters on a while background, thus simulating the appearance of an index card. When this inverse-video data is dumped to the NEC 8023, the printer shifts to a Greek character set that is incomprehensible to most mere mortals. Again,

continued on next page

the key to solving this problem rests with our old friends, the DIP switches. By changing Switch 2, Position 6 from the factory-set 8-bit to the 7-bit mode, you can set the printer to accept 7-bit data. The inverse indicator apparently is located in the dropped eighth bit. So we've "tricked" the printer into thinking the inverse characters are actually garden-variety ASCII non-inverse characters. This change does not affect most other operations, but graphics operations normally do require the factory-set, 8-bit position.

However, the NEC printer is more likely to be used for word processing applications than for graphics printing, and I've found many word processors to be either unworkable or awkward to use with the NEC/Atari combination. One that works very well, though, is the new AtariWriter. For the most part, a user can follow the instructions supplied with the AtariWriter program and select the Atari 1025 mode for printing with satisfactory results. To get maximum use from the program, however, one must go beyond the vague references in the manual. Most typefaces and special features can be accessed through the printer-control feature of the program, rather than through its print-style mode and other, similar modes.

By pressing the [CTRL] key, followed by the letter O, it is possible to insert control codes that the NEC printer will understand. For example, if you type [CTRL] O 27 [CTRL] O 80, the printer will print all text that follows in the proportional mode. (The proportional mode can also be set as the default font, replacing 10 CPI, by changing the appropriate DIP switch.)

Other examples follow:

TYPE THIS	GET FUNCTION
[CTRL] O 27 [CTRL] O 78	Pica/10CPI font
[CTRL] O 27 [CTRL] O 81	Condensed/17CPI font
[CTRL] O 27 [CTRL] O 69	Elite/12CPI font
[CTRL] O 27 [CTRL] O 88	Begin underlining
[CTRL] O 27 [CTRL] O 89	End underlining
[CTRL] O 14	Being elongate
[CTRL] O 15	End elongate

Rather than list every possible code for every possible function, I'll refer you to the appropriate reference materials. The NEC 8023 manual describes all of the functions of the DIP switches, escape codes and other control codes. "Appendix C" in the *Atari BASIC Reference Manual* equates the escape-code characters, character-string hexadecimals, and decimal numbers. Using this chart, together with the NEC manual (and a little imagination), you should be able to manipulate the NEC 8023 printer by using any software that accepts printer controls, as well as by using BASIC's LPRINT statements and character strings. The NBASIC in the NEC manual is not identical to Atari BASIC, but I've discovered that a little experimentation will take you a long way.

ARCADE CLASSICS. A BLAST-FROM-THE-PAST.



Remember those great arcade games of the past?

Games like Bally Midway's™ Seawolf II.™ As sub commander, you torpedo enemy ships moving at different speeds. Sink as many as you can, but watch out for the mine fields.

And Bally Midway's Gun Fight.™ In the most famous shootout of all time, you'd better be a straight shooter and fast on the draw.

Exidy's™ Starfire™ gives you outerspace graphics that are out of this world. Plus a starship with laser cannons to help you rid yourself of alien forces.

And there's Exidy's Fire One!™ Hone in on the enemy with sonar. Then it's up periscope and torpedoes away.

EPYX has brought the arcade classics home—at the right price for you. We've combined the Bally Midway classics Seawolf II and Gun Fight on one disk and Exidy's classics Starfire and Fire One on another. So get the arcade classics today and have a blast-from-the-past on us.

One or two players; joystick controlled; arcade style graphics and sound.

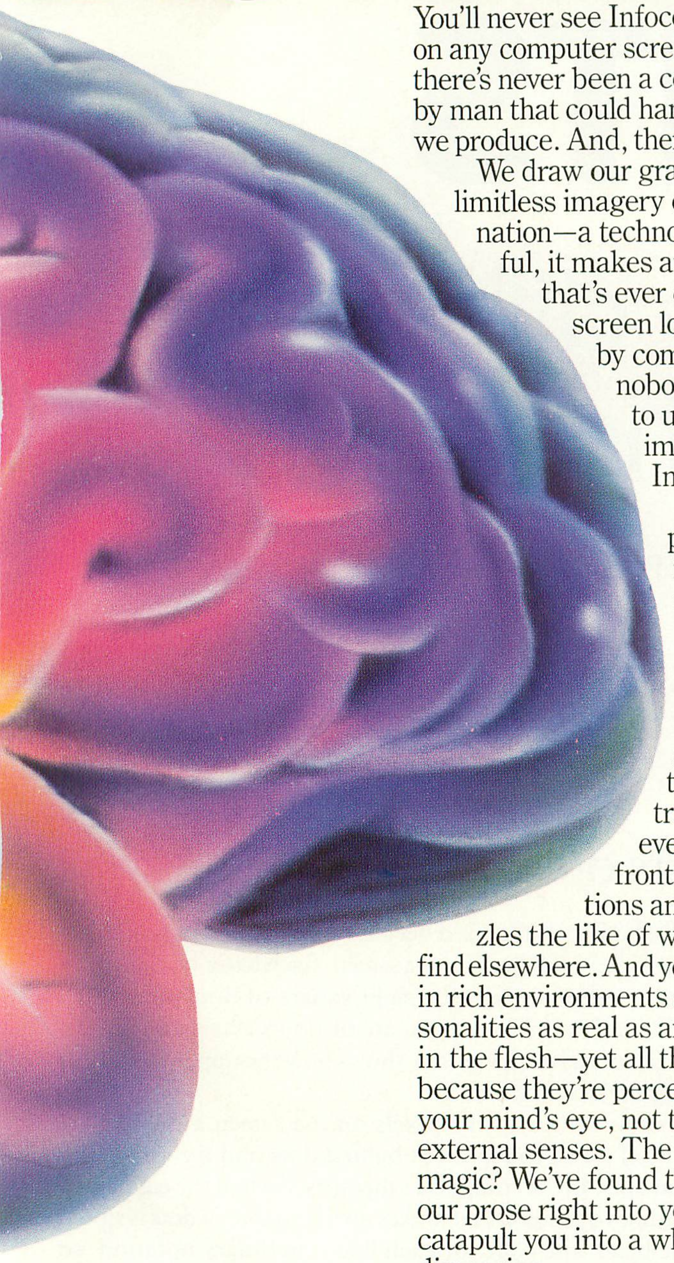


EPYX™ COMPUTER SOFTWARE

Strategy Games for the Action-Game Player



E WORLD'S MOST
HICS TECHNOLOGY.



You'll never see Infocom's graphics on any computer screen. Because there's never been a computer built by man that could handle the images we produce. And, there never will be.

We draw our graphics from the limitless imagery of your imagination—a technology so powerful, it makes any picture that's ever come out of a screen look like graffiti by comparison. And nobody knows how to unleash your imagination like Infocom.

Through our prose, your imagination makes you part of our stories, in control of what you do and where you go—yet unable to predict or control the course of events. You're confronted with situations and logical puzzles the like of which you won't find elsewhere. And you're immersed in rich environments alive with personalities as real as any you'll meet in the flesh—yet all the more vivid because they're perceived directly by your mind's eye, not through your external senses. The method to this magic? We've found the way to plug our prose right into your psyche, and catapult you into a whole new dimension.

Take some tough critics' words about our words. SOFTALK, for example, called ZORK® III's prose "far more graphic than any depiction yet achieved by an adventure with graphics." And the NEW YORK TIMES saw fit to print that our DEADLINE™ is "an amazing feat of programming." Even a journal as video-oriented as ELECTRONIC GAMES found Infocom prose to be such an eye-opener, they named one of our games their Best Adventure of 1983.

Better still, bring an Infocom game home with you. Discover firsthand why thousands upon thousands of discriminating game players keep turning everything we write into instantaneous bestsellers.

Step up to Infocom. All words. No graffiti. The secret reaches of your mind are beckoning. A whole new dimension is in there waiting for you.

(For more information on Infocom games contact: Infocom, Inc., P.O. Box 855, Garden City, NY 11530.)



INFOCOM™

The next dimension.

For your: Apple II, Atari, Commodore 64, CP/M 8,* DEC Rainbow, DEC RT-11, IBM, MS-DOS 2.0, NEC APC, NEC PC-8000, Osborne, TI Professional, TI 99/4A, TRS-80 Model I, TRS-80 Model III.

ROTATING FONTS

Printing with different fonts on the NEC

by SOL GUBER

Having purchased a printer recently, I was very anxious to use it as soon as possible. The printer itself worked well, but because I had designed custom fonts for some of my programs (using the common technique of character graphics) a few problems developed. I discovered that my printer, a NEC 8023A-C, could normally only print English or Japanese characters (since I don't know Japanese, this wasn't much of a selling point). The following program was designed to get around this problem. If your computer can use a particular font, this program will allow you to print it out on your printer, as long as it has an individually-addressable pin configuration (such as the Epson with Graftrax or the NEC).

The program is set up specifically for the NEC 8023A-C printer; modifications are needed to make it work for other printers. It has two parts. The first is a utility that rotates the letters in the present character font; the second, also a utility, reads the characters on the screen in Graphics 0 notation and sends them to the printer using dot-matrix notation. I'll deal with each of these utilities individually.

DOTS ON THE SCREEN

The first utility is required because of the different structures of the printer and the monitor screen. Each letter on the screen is composed of an eight-by-eight square of dots (or pixels). When the dots in the square are lit up in a certain pattern, your eye sees a letter. Figure 1 shows how the upper-case letter "A" appears on the screen.

To show which of the pixels on the screen are lighted, I have used a one to represent lighted dots and a zero for unlighted dots. In the computer's memory, each of the eight horizontal lines of coding that makes up the square of dots is stored in one byte. The code for each line is in binary notation, so that 00011000, for example, equals 24. Each of the pixels is so small that when all eight lines are seen together they appear to form a continuous letter on the screen against the darker background.

This same kind of information can be transmitted to the printer and used to generate a printed letter, if the proper dot-image graphics mode is selected. The NEC 8023A-C features

a dot-matrix printing mode that can be selected by using the "ESC-S" code and a four-digit number that signifies how many bytes of information are to be transferred.

HORIZONTAL VS. VERTICAL

However, there is one problem that needs to be resolved before this information can be transferred to the printer. On the screen, the byte information is read horizontally (that is, each line of eight dots (see Figure 1) is read from side to side across the screen. But the pins on the NEC printer and on any pin-addressable printer are vertical. As a result, the same information that is read correctly on the screen will appear to be "sideways" to the printer.

Of the eight vertical pins on the NEC printer, the least significant bit (LSB) is the top pin, and the most significant bit (MSB) is the bottom pin. Thus, the data "1" will cause the top pin to print a dot, while the data "128" will prompt the bottom pin to do so.

binary	hexadecimal	decimal
00000000	00	0
00011000	18	24
00111100	3C	60
01100110	66	102
01100110	66	102
01100110	66	102
01111110	77	121
01100110	66	102

Figure 1

A ROTATING SUBROUTINE

Because the information that the printer needs is different in form from that the computer itself uses, a subroutine had to be written to "rotate" it so that the printer could print the characters correctly. It is possible to use Graphics Mode 8 to create characters that are sideways or upside-down, and in this case something similar must be done to modify the letters in the computer's memory. Figure 2 shows an example of this.

binary	hexadecimal	decimal
00000000	00	0
11111000	F8	248
11111100	FC	252
01000110	46	70
01000110	46	70
11111100	FC	252
11111000	F8	248
00000000	00	0

Figure 2

Memory location 756 contains a pointer that tells the computer where to find the character set that is to be used. This pointer can be used to modify the letters in the character set, and we will do so to create our modified font.

The subroutine in lines 1000–1080 is used to rotate the present font to make it conform to the needs of the printer. Line 1000 determines the address of the present font, which is stored in memory location 756. Many calculations will be needed at this point, so we need to turn the screen off. To do this, POKE 0 to location 559.

Next, a loop for all of the letters (taken a byte at a time) is started. In line 1110, two variables are initialized. These are the low bit in the byte (N) and the current offset within the eight bytes that makes up the letter (U1).

Line 1120 determines the value in the memory location. KK is the bit that will be placed in each byte if a one is found there. Line 1125 sets the memory location of the new font, and includes a pointer to the exact spot where the new font is located.

Line 1130 determines if there is a one in the rightmost bit of the byte used for a letter. If there is a one present, N is added to the value already there. Line 1140 then removes the rightmost bit and decrements KK. If all eight bits in the byte have not yet been processed, lines 1130–1140 need to be repeated. If the byte has been finished, line 1150 doubles the value in N and increments U.

If all eight bytes for a letter have not yet been processed, the next byte in the letter needs to be handled in this manner. If the letter is finished, process the next letter. If all 128 characters have been processed, turn the screen back on by POKEing 559 with 34, and return to the main program. This program takes about three minutes to work.

TRANSMITTING TO THE PRINTER

The second part of this program, also a utility, uses information in screen memory to transmit information to the printer. The method it uses is similar to that used by the Operating System (OS), and it saves several steps because of this.

The Atari system uses a portion of memory as the location for the screen memory. The information that is stored there is put onto the screen by the ANTIC chip. This area can be checked to determine what is on the screen, and can be directly modified without going through PRINT statements. A pointer at locations 88 and 89 shows the beginning of screen memory. And locations 87 tells ANTIC which of the BASIC graphics modes has been used to code the information in screen memory. As a result, the same memory information can be used to signify letters — as well as colors — on the screen, if the proper decoding scheme is used.

The OS uses the pointer to locate the first byte of screen memory. It then takes this information and decides how to decode it. Since we're using Graphics Mode 0, the decoding is rather straightforward. The OS simply takes the value in the byte and multiplies it by eight. It then adds this value to the start of the character memory to find the next eight bytes of information. These will be used to determine the shape and form of the letter. (The OS then determines where to put these eight bytes on the screen, but I haven't yet discovered what system it uses to do this.)

MIMICKING THE OS

My second utility uses a similar method for sending information to the printer. It determines the beginning of screen memory and puts this number at the start of a FOR/NEXT loop. The program then finds the value of the screen memory. Next, it multiplies this value by eight and adds the product to the beginning of the NEWFONT memory. Then the program takes the next eight bytes and puts them into a string that will output this information to the printer. This is done forty times, for the forty letters on the screen. Then it goes on to the next line. After 23 lines, the subroutine returns to the major portion of the program.

continued on next page

PROGRAM DETAILS

Let me go through the main portion of the program now in more detail. Lines 1–5 initiate some of the variables that will be used in the program (the use of variables, rather than constants, speeds the program up significantly). Line 20 dimensions FONT$. This is a space in which to put the rotated font. It is needed as a backup in case of an accident (such as a change in the graphics mode) so that the information will not be lost. NFONT is the address in memory for the start of FONT$. It's used in the rotation subroutine.

Line 30 zeroes FONT$. Line 95 goes to the rotation subroutine. Line 100 dimensions the string used to output to the printer. Line 120 determines the start of screen memory. Line 200 lists part of the program so that you will have something to print out on the printer (as a test of the program).

Line 330 sets K to the location of the first spot of screen memory. Line 350 contains the FOR/NEXT loop that gets the value at the memory location (CHAR) and puts it into the string OUT$ in the proper place. The program determines which information to use by multiplying the value in CHAR by eight and using the next eight bytes in FONT$. Line 350 prints the new value using the NEC's proper dot-matrix notation. Then line 500 determines if all 23 lines have been checked and if the system is finished. If so, the program stops.

binary	hexadecimal	decimal
00000000	00	0
00011111	1F	31
00111111	3F	63
01100010	62	98
01100010	62	98
00111111	3F	63
00011111	1F	31
00000000	00	0

Figure 3

MODIFICATIONS FOR OTHER PRINTERS

This program should work on any printer with addressable pins that handles graphics output the way the NEC does. For printers that use a pin configuration that is the reverse of the NEC's, however, the program will require modification. These printers require bytes in memory (for each letter) that look like the configuration in Figure 3. This necessitates two changes to the program. Lines 1110–1150 should be changed to the following for use with such printers:

```
1110 N = C128:U = C0
1120 AB = PEEK(OLDFONT + J + U):KK = C0
1125 SP = NFONT + J + KK
1130 IF AB/C2 <> INT(AB/C2) THEN POKE SP,PEEK
     (SP) + N
1140 AB = INT(AB/C2):KK = KK + C1: IF KK>C7
     THEN GOTO C1125
1150 N = N/C2:U = U + C1: IF U<C8 THEN GOTO
     C1120
```

When a different font is used, this program may need to be modified so that the letters are enlarged, because there is more spacing on a printer than on the screen.

```
1 REM *ROTATING FONTS*
2 REM * BY SOL GUBER *
3 REM *ANTIC MAGAZINE*
4 REM *JANUARY, 1984 *
5 REM ****************
10 CØ=Ø:C1=1:C2=2:C3=3:C4=4:C5=5:C6=6:C7=7:C8=8:C9=9:C1Ø=1Ø:C4Ø=4Ø:C128=128:C256=256
15 C112Ø=112Ø:C1125=1125:C39=39:C35Ø=35Ø:C4Ø=4Ø
2Ø DIM FONT$(1Ø9Ø):NFONT=ADR(FONT$)
3Ø FONT$(C1)=CHR$(CØ):FONT$(1Ø3Ø)=CHR$(CØ):FONT$(C2)=FONT$
95 GOSUB 1ØØØ
1ØØ DIM OUT$(32Ø)
1Ø5 OUT$(1)="♥":OUT$(32Ø)="♥":OUT$(2)=OUT$
11Ø OUTADR=ADR(OUT$):NECHAR=PEEK(756)*256
12Ø SCREEN=PEEK(88)+256*PEEK(89)
2ØØ LIST 1ØØØ,2ØØØ
3ØØ REM
33Ø K=SCREEN
35Ø FOR DOT=CØ TO C39:CHAR=PEEK(K+DOT):OUT$(DOT*C8+C1,DOT*C8+C8)=FONT$(CHAR*C8+C1,CHAR*C8+C8):NEXT DOT
45Ø LPRINT "␛SØ32Ø";OUT$;"      "
5ØØ IF K<SCREEN+88Ø THEN K=K+C4Ø:GOTO C35Ø
999 STOP
1ØØØ OLDFONT=PEEK(756)*256:POKE 559,Ø:FOR J=CØ TO 1Ø27 STEP C8
111Ø N=C1:U=CØ
112Ø AB=PEEK(OLDFONT+J+U):KK=C7
1125 SP=NFONT+J+KK
113Ø IF AB/C2<>INT(AB/C2) THEN POKE SP,PEEK(SP)+N
114Ø AB=INT(AB/C2):KK=KK-C1:IF KK>CØ THEN GOTO C1125
115Ø N=N*C2:U=U+C1:IF U<C8 THEN GOTO C112Ø
118Ø NEXT J:POKE 559,34:RETURN
```

TYPO TABLE

Variable checksum = 1Ø37426

Line num range	Code	Length
1 - 1Ø5	MJ	492
11Ø - 112Ø	LI	321
1125 - 118Ø	OD	129

DOING THE DIP-SWITCH DOODLE

Printer control begins here

by ROBERT DEWITT
Managing Editor

Almost every printer has DIP switches for controlling the mysterious functions that are described (if you're lucky) in the back of the printer's instruction manual. These switches are usually tucked away in some inaccessible recess, which suggests that the manufacturer really doesn't want you messing with them. Unfortunately, you often must find and reset the DIP switches to get your printer to work properly with your computer and software.

DIP is an acronym deriving from "dual in-line package," which is a fancy way to say that several miniature ON-OFF switches are packaged together in a row. Usually each switch in each package is numbered; if two or more packages of switches are used together, each package is numbered. For example: SW2-3 would designate the third switch in the second package. This code is essential to finding the correct switches. The ON position is usually indicated, but if not, you will have to experiment.

Always turn off your printer and computer when setting DIP switches. Serious damage can occur if power is improperly shunted within the printer circuitry. Besides, you might get a nasty shock.

Most companies ship their printers with the DIP switches preset to the standard, or default, arrangement for most buyers. If your printer doesn't work, or won't do everything you think it should, check out the DIP switches.

DIP switches are small and sometimes you must partially disassemble your printer to get at them. But first, look carefully for a little door or pop-off panel. If you disassemble your printer, you run the risk of dislodging connectors or misaligning mechanical parts.

A very important and often overlooked aspect of DIP switches is that resetting them has no effect until the printer has been turned off and then on again. Changing a DIP switch while a printer is in operation, which is not advised, will not invoke the functional change desired.

What kind of functions are controlled by DIP switches in printers? Here are some of the common ones:

- Number of lines between tops of forms (usually 66, but switchable to 72). This is also called "skip over perf."
- Command sequence for new line. Will the computer send a carriage return only, or CR plus other codes such as line feed? The Atari sends CR only, so this DIP is always important for us.
- Changing character sets, for example to Japanese, Greek, or custom-designed characters (if printer allows this).
- Enable select-deselect, i.e. on-line or off-line states while printer is in power-on condition.
- Font mode changes (enhanced, regular or compressed characters). You can usually do this with software commands as well.
- Zero font ON will print zeros with a slash through them to differentiate from letter O.
- Buffer on or off. Some printers have sizable internal buffers.

Every set of DIP switches is different, and even successive models of the same brand of printer may have different settings. There is no alternative to reading your manual and learning how your computer and software send codes to the printer. While this relationship is complex and frustrating to master, your reward will be to have confident control over these expensive tools you have chosen to have in your life.

PRINTER SURVEY

Sooner or later every computer owner wants a printer. Printers are expensive, and each make and model is subtly different from the others. It's probably as hard to pick the right printer as it is to pick the right computer. This survey examines a number of printers suitable for use with the Atari computers. We hope it is helpful to you.

by ROBERT DEWITT
Managing Editor



ALPHACOM 42
Alphacom
2323 So. Bascom
Campbell, CA 95008
(800) 227-6703
In CA (800) 632-7979

The Alphacom 42 is a small utility printer that offers good value — if you can accept its limitations. It is especially useful for printing listings, and is capable of graphics if you can do a little programming. It costs only $99.95, complete with cable.

The main limitation of the printer is the 4½" width of the paper it uses, which comes on rolls that will hold about 10,500 lines of print. This is just the right width to present lines of text or code as they appear on the video display. The printer is "thermal," which means that the image is created by a heated element. This is the same technology that has worked reliably and safely in Olivetti calculators for many years. However, the thermal paper has been treated with chemicals, and some people may be allergic to it. Two types of paper are available — blue image or black image. Blue-image paper will not xerox well. The paper costs about $3.00 per roll.

The appearance of the letters produced by the Alphacom is quite acceptable, and letters requiring descenders are clearly readable — a quality not always found in inexpensive printers.

The happiest feature of the Alphacom 42 is its ability to depict the complete Atari character set, including graphics characters, accurately in both normal and inverse fashion. This is accomplished by incorporating a special ROM in the interface cartridge. Many printers either cannot print Atari graphics characters at all, or can do so only with special software in a graphics mode.

The Alphacom is fast enough for most purposes. We clocked it at about two lines (80 characters) per second, and it is quiet. This is not a printer suited to business purposes, nor to word processing. Most word processors rely on non-printing control characters as formatting commands, but the Alphacom prints *any* control character for which Atari specifies an image. You may be able to program around this problem, but the

continued on next page

Alphacom manual says nothing about it. If document size or shape must be standard, you are also out of luck with the Alphacom.

On the plus side, the Alphacom sets up easily and goes right to work without much fuss. The Atari cable is simple to connect, and the instructions are adequate for any patient user. The printed product is legible and handy for many purposes — including casual correspondence.



ATARI 1027

Atari, Inc.
1265 Borregas Ave.
Sunnyvale, CA 94086
(800) 538-8543 or
(800) 672-1404 in CA

A new printer on the scene is the Atari 1027, a letter-quality printer that costs less than $350. It is designed to work directly with Atari computers, and does not require an interface or unusual cable. Cartridge-based word processors like AtariWriter can produce documents in RAM and dump them to the 1027 without using a disk drive or cassette recorder. Of course, this means you will have no copy of your work on media, but a hardcopy record is satisfactory for many purposes, such as correspondence and schoolwork.

The 1027's elite (12-pitch) letters and numbers look like the typical typewriter's characters, which is very desirable when the recipient of the document may react unfavorably to dot-matrix characters. Control codes can elicit a variety of accent marks that are common in European languages. This thoughtful feature enhances the machine's use as a school tool.

Operation is simple. The paper slot accommodates single sheets up to 8½" wide. Fan-fold roll paper can be used if the tractor holes are stripped off, but this printer is really made for friction-fed single sheets.

PRINTER MODEL	ALPHACOM	ATARI 1027	AXIOM AT-100	EPSON FX-80
PRINTER TYPE	THERMAL	LETTER QUALITY	DOT MATRIX	DOT MATRIX
RETAIL PRICE	$99	$349	$299	$699
PAPER TYPE	THERMAL ROLL	SINGLE SHEET[4]	FAN FOLD	FAN, SINGLE
SIZE	4½"	8½"	4½"-10"	9"-10½"
FRICTION FEED	YES	YES	NO	YES
TRACTOR FEED	NO	NO	YES	YES[12]
MINIMUM PAPER ADVANCE	?	⅙"	1/18"	1/216"
FRONT PANEL CONTROLS	POWER PAPER ADVANCE	POWER	ERROR LAMP	LF, TOF SELECT PAPER EMPTY[9]
DIP SWITCH ACCESS	NO	NO	DIFFICULT	EASY
RATED CHAR. PER SEC.	80	?	30	160
BENCHMARK* CHAR. PER SEC.	80	8	17	69
BENCHMARK TIME	—	550 sec.	235 sec.	58 sec.
CHARACTERS PER INCH	10	12	5 & 10	5,10,12,16[10]
PROPORTIONAL FONT	NO	NO	NO	YES
DESCENDERS	1 dot	TRUE	NO	2 dot
SUPERSCRIPT/ SUBSCRIPT	NO	NO	NO	YES
OVERSTRIKE	NO	NO	NO	YES
UNDERLINE	NO	YES	NO	YES
SELF TEST	YES	NO	DIFFICULT[2]	YES
BIDIRECTIONAL	NO	YES[5]	NO	YES
SHORT LINE LOGIC	NO	YES[5]	NO	YES
GRAPHICS CAPABLE	YES[1]	NO	YES[3]	YES
RESOLUTION (DOTS VERT/HORIZ)	?	NONE	63x60	40x240[11]
# OF PINS IN PRINT HEAD	?	NOT APPLICABLE	7	9
RIBBON TYPE	NONE	PAD	FABRIC CART.	FABRIC CART.
COST	NONE	$2.20	$9.95	$9.00
ATARI CABLE PROVIDED	YES	YES	YES	NO
BUFFER SIZE	NONE	NONE	90	80
850 INTERFACE REQUIRED	NO	NO	NO	YES
INSTRUCTION MANUAL	ADEQUATE	ADEQUATE	POOR	EXCELLENT

1 Atari character graphics only
2 Instructions inadequate
3 GraphAX costs $15 extra
4 Fan fold okay if tractor holes stripped
5 Only on last line of paragraph
6 Friction feed costs $150 extra

*The benchmark test we devised for these printers consisted of approximately 4,000 characters of English text in a Letter Perfect file.

The mechanism for typing resurrects an old technology that was used on teletype machines. Five wheels on a spindle carry the character faces. These spin rapidly over the paper. When the printer receives the code to print a letter, the mechanism positions the correct wheel at the next spot to be typed, stops the wheel at the correct letter, presses it against the paper and returns to print the next letter. There is no ribbon. All letters are inked continuously from a pad.

This technique is not as elegant as a daisywheel, but it's cheaper and it works. However, one of its problems is speed. In our benchtest, the 1027 required more than nine minutes to type the copy that most dot-matrix printers printed in one minute. Its durability is also questionable. This mechanism

EPSON RX-80	GEMINI-10X	MANNESMANN-TALLY SPIRIT	NEC PC-8023A	TRANSTAR 120
DOT MATRIX	DOT MATRIX	DOT MATRIX	DOT MATRIX	LETTER QUALITY
$299	$399	$399	$645	$599
FAN FOLD	FAN, SINGLE, ROLL[7]	FAN, SINGLE	FAN, SINGLE	FAN, SINGLE
4½"-10"	2½"-10"	4"-10"	4½"-10"	0-13"
NO[6]	YES	YES	YES	YES
YES	YES	YES	YES	NO
?	1/144"	1/216"	1/144"	1/48"
LINE FEED TOP OF FORM SELECT	LINE FEED TOF-SELECT PAPER EMPTY	LINE FEED TOF-SELECT PAPER EMPTY[9]	LF, TOF SELECT PAPER EMPTY	LINE FEED AUTOLOAD SELECT, ALERT
?	DIFFICULT	DIFFICULT	EASY	EASY
100	120	80	120	14
55	59	53	61	12
73 sec.	68 sec.	75 sec	66 sec	340 sec.
5,10,12,16	5,10,12,16	5,10,16	5,10,12,16	10,12,15
NO	NO	NO	YES	NO
2 dot	2 dot	2 dot	2 dot	TRUE
YES	YES	YES	NO	NO
YES	YES	YES	YES	YES
YES	YES	YES	YES	YES
NOT TESTED	YES	YES	YES	YES
YES	YES	YES	YES	NO
YES	YES	YES	YES	NO
YES	YES	YES	YES	NO
?	144x240	160x216	160x144	NONE
9	9	9	9	NOT APPLICABLE
FABRIC CART.	FABRIC SPOOL	FILM OR FABRIC	FABRIC CART.	FILM OR FABRIC
$9.00	$2.49	CART $8.95	2 for $25.00	CART $6.50
NO	NO	NO	NO	NO
NONE	NONE	90	2K	NONE
YES	YES	YES	YES	YES
ADEQUATE	GOOD[8]	ADEQUATE	ADEQUATE	GOOD

7 Roll holder provided
8 List of control codes provided
9 Audible "paper-empty" alarm
10 Custom fonts from computer also will print
11 Maintains true ratio of dimensions
12 Adjustable tractors cost extra

looks and sounds like it will go berserk at any minute. In fairness, though, we should report that it did not fail during our brief experience with it.

The 1027 matches the new XL design, and will look nice with any Atari gear. It is a reasonable choice for limited uses in which speed is not important, but the appearance of the output is.

AXIOM AT-100 (SEIKOSHA)

Axiom Corp.
1014 Griswold Ave.
San Fernando, CA 91340
(800) 232-9466
In CA (213) 365-9521

It's easy to find fault with the Axiom AT-100 dot-matrix printer (made by Seikosha) — it's slow and noisy, it only uses computer paper, and its only font has no proper descenders. But it does have some redeeming virtues. It only costs $299, it is easy to use, and the built-in cable that connects it to Atari computers bypasses the expensive and hard-to-find 850 Interface.

There is room for this printer in the market, and it makes a good "starter" printer for the student or hobbyist. The print quality it provides will have limited acceptability for most readers, however,



so give some thought to the probable uses of your printouts before you buy.

The AT-100 is really a Seikosha GP-100A printer that has been specially configured for Atari systems. The same printer is marketed as the Gorilla Banana, the Panasonic JR-PO2U, and in versions specific for the Commodore and Spectravideo computers. It is a very popular device in all its guises, but these other versions do not work directly with the Atari as does the AT-100. Further, Axiom Corp. offers a GraphAX package for $15 that substantially improves the graphics ability of the AT-100 for Atari owners. Even without GraphAX though, you can still print screen images of text and high-resolution drawings by using BASIC screen-dump routines furnished by Axiom.

The AT-100 does not take single-sheet paper such as letterhead, but does accept computer stock as narrow as 4½" labels, for example. Its only control is the power switch at the rear. Two lights on the front indicate POWER and ERROR. The normal font provides 10 characters per inch, but enhanced (double-wide) characters can be invoked through software commands.

One big drawback with this printer is its speed. It is said to print 30 characters per second, but required 235 seconds to print our benchmark material of 4000 characters (about 17 characters per

continued on next page

second). The discrepancy is apparently accounted for by its carriage return time of about one second per line. Many more-expensive printers are not only faster in absolute terms, but they also print in both directions, thereby saving the return time.

Paper feeds into the printer forward only, and damage could occur if you attempted to pull it out backwards. This is not a major problem, but it does mean that you must waste a sheet of paper between documents to set the next job up properly.

The generic instruction manual for the Seikosha GP-100A is supplied with the printer, and is atrocious. Luckily, Axiom has added several decently-written pages that will help you learn to operate this essentially simple machine.



EPSON FX-80

Epson America
3415 Kashiwa St.
Torrance, CA 90505
(800) 421-5426 or
(213) 539-9140 in CA

The Epson FX-80 with Graftrax Plus is probably the most coveted dot-matrix printer in computerdom. Epson developed the features that others in the industry emulate, and the company is still the leader in high-resolution graphics printing for the microcomputer market.

The Epson offers nicely-formed characters in regular and italic fonts, single strike or doublestrike (boldface) formats, and in three different piches: 10-pitch (pica), 12-pitch (elite) and 16-pitch (compressed). Each can be expanded to double width, so it is possible to have combinations like "doublestrike expanded-compressed italic" letters that are also underlined and subscripted. You can also download your own customized character set and print with it. Sanskrit, anyone?

As a graphics printer, the Epson excels, featuring a dot density as high as 240 × 240 per square inch. Graphics are usually accomplished with seven or eight pins, depending on the computer originating the signal. The Epson can use all nine pins at once to print graphics, and this can speed up the process significantly. Graphics-utility programs are almost always written to work with Epson's Graftrax, and with other printers if you're lucky. It is the fastest printer we tested — rated at 160 characters per second and benchtested at 69 CPS.

Controls on the FX-80 include line feed, top of form and select. A paper-out sensor stops operation, lights a warning lamp and rings a bell. The FX-80 prints bidirectionally and uses short-line logic. A buffer holds about one line of text.

The Epson User's Manual is an excellent support tool — really a course in computerized printing. It is written for the non-technical owner and is filled with examples and exercises. It virtually assures the motivated owner of full access to this printer's many features.



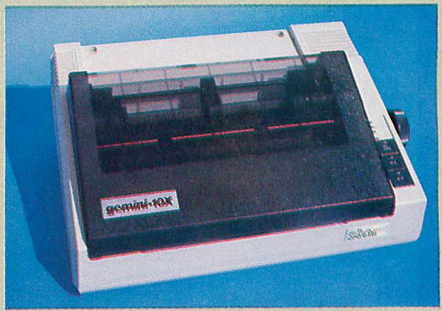
EPSON RX-80

Epson America
3415 Kashiwa St.
Torrance, CA 90505
(800) 421-5426 or (213) 539-9140

This little printer just might run away from the field. An inexpensive unit from an industry leader, this dot-matrix machine offers quality and features previously available only on much more expensive equipment.

The RX-80 draws on proven Epson MX-series technology and introduces several new functions, including real italics that can be mixed with its other text styles. Its most serious drawback is that it is limited to tractor-fed computer paper. Friction feed of single sheets (such as letterhead) is available for $150 extra, but at that price other printers are equally attractive. The cable also costs extra.

The printhead has nine pins (seven is more common at this price), and the appearance of the letters it produces is quite suitable for all but the most formal needs. Schoolwork, for example, should certainly be complemented by the subscripts, superscripts, italics and underlining made possible by the RX-80. It is rated at 100 characters per second, but it tested a bit slower in action — 55 CPS. Still, that's plenty fast enough for most uses.



GEMINI 10X

Star Micronics
#3 Oldfield
Irvine, CA 92714
(714) 768-4340

This is an Epson lookalike, and a good buy in the medium-priced dot-matrix market. It accepts any kind of paper, including rolls (for which it provides the rigging), and has friction feed along with fully-adjustable tractor feed, down to two and a half inches.

The Gemini 10X is rather noisy, and tested significantly slower than its rated 128 characters per second. We got only 59 CPS on our benchtest, but that included carriage returns and line feeds. However, 59 CPS is not especially slow in comparison to other printers tested in the survey.

The standard fonts (wide, pica, elite and condensed) are available in a nicely-designed dot matrix with true descenders. The pica face looks very "dotty," due possibly to the very fine points of the pins. While these contribute to a fine graphics presentation, the pica font probably would not be good for business correspondence.

Two features of the Gemini are especially attractive: It has a "paper-out" signal lamp, and it automatically goes off-line if it runs out of paper. Printing

continued on page 113

CASADAPTER
"IT REALLY WORKS!"

CASADAPTER is a cassette interface that allows you use your own cassette recorder or stereo with the Atari 400/800/1200®. **CASADAPTER** will handle motor control, audio and data channels.

$34.95

48K RAM BOARD FOR: THE ATARI 400®.

$115.00

***MAGIC DUMP** is a screen dump utility that allows you to dump a Hi-Resolution graphics picture to a printer in a variety of different sizes. **MAGIC DUMP** is used in the RIGHT hand cartridge slot, so it is always ready to use.

MAGIC DUMP will work with all Epson printers and Gemini printers, the Centronics 739 printer, and the Prowriter or N.E.C. printers. **$59.95**

***GTIA DRAW** is a drawing program that uses all the features of Atari's® new GTIA chip. **GTIA DRAW** will give you three extra graphics modes, Mode 9 (16 luminances and one color), Mode 10 (8 luminances and color), or Mode 11 (16 colors and one luminance).

GTIA DRAW will allow you to:

- Add text to pictures
- Blank horizontal or vertical lines
- Shift the screen in any direction
- 'ZOOM' in on certain areas of a picture
- Fill portions of a screen
- Work on two different screens simultaneously

$49.95

***THE DRUMESISER** is a unique sound synthesizing tool that allows the creation of sounds such as a drum, piano, organ, harpsichord, or electronic synthesizers. The different types of sounds created are limited by the user's imagination and ambition.

THE DRUMESISER comes with an Editor, which allows you to create your own sounds, a Player, which will allow you to play the different instruments, and the Memory Options, which make it possible for you to playback any sounds that you have recorded.

$49.95

SAR-AN COMPUTER PRODUCTS

12 Scamridge Curve
Buffalo, New York 14221
(716) 632-3441

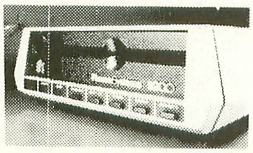
Dealer Inquiries Invited
*Product of Gemini Software

Add $2.50 shipping.
Send check or money order
C.O.D. accepted
New York State Residents add 7% tax
®Atari is a trademark of Atari, Inc.

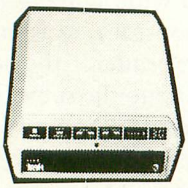
A LIFETIME OF DISCOVERY BEGINS WITH ATARI COMPUTERS

ATARI

RANA 1000 DISK DRIVE



Single/Double Density Drive ....
Atari DOS .... CALL
Double Density Software ......



TRAK AT DISK DRIVE

Single Density AT-D1 .... CALL
Double Density AT-D2 .... FOR
Atari DOS .... PRICES
Double Density Software ......

PRINTERS

Gemini STX-80 (80 Column Thermal)	$169.00
Gemini 10X Printer (80 Column)	299.00
Gemini 15 Printer (136 Column)	399.00
Silver-Reed EXP 500 Daisy Wheel Printer (80 Column)	399.00
Epson RX-80 (80 Column) Dot Matrix	359.00
Epson FX-80 (80 Column) Dot Matrix	555.00
Epson FX-100 (136 Column) Dot Matrix	749.00
Prowriter 8510 AP	375.00
Gorilla/Banana Printer (80 Column)	199.00
MPP-1100 Parallel Printer Interface (Eliminate need for Atari 850)	79.00

MONITORS

Gorilla™ Hi Res 12" Non-Glare Green Screen	$ 89.00
Amdek Video 300 Green Screen	130.00
Amdek Video 300A Amber Screen	144.00
Amdek Color-I Screen	280.00
Monitor Cable	15.00

MODEMS

MPP-1000 Modem	$169.00
Signalman Mark II Modem	69.00

600 XL	C	1050	Drive
800 XL	CALL	1025	Printer
1400XL	L	1020	Color Printer
1450XL	L	1027	Printer
		1010	Recorder

ATARI - 3rd Party Software
ALL BRANDS - CALL

GENERIC DISKS

GENERIC 100% DEFECT-FREE/GUARANTEED MINI-FLOPPY DISKS
Diskettes (1 Box Min.) - 10 per box

	SS/SD	SS/DD	DD/DD
1 or 2 Boxes	14.99/box	17.49/box	20.99
3 - 9 Boxes	13.99/box	15.99/box	19.99
10+ Boxes	13.49/box	14.99/box	18.99

Bulk Diskettes with Sleeves - Price Per Disk

	SS/SD	SS/DD	DD/DD
10 - 29	1.39	1.59	1.99
30 - 99	1.29	1.49	1.89
100+	1.25	1.45	1.79

TO ORDER CALL TOLL FREE: 1-800-824-7506

MC
VISA
C.O.D.
(Add $2.50)

COMPUTER CREATIONS, Inc.
P.O. Box 292467
Dayton, Ohio 45429
For Information Call: (513) 294-2002
(Or to order in Ohio)

All Orders Add $3.00 Shipping and Handling. Ohio residents add 6% for sales tax.

ASK FOR OUR FREE CATALOG

DISK LABEL PRINTER

Epson/Graftrax Plus packs them in

by RICHARD KUSHNER

"Where did I put that program I just modified?" I asked myself.

I changed some of the statements to improve the performance and appearance of the output, saved it on disk and went away to do something else. Back at the computer, I was ready to use it and — it was lost! Thirty minutes later, after using DOS to search the directories of 40 disks, I found it.

"Is this any way to spend what little time I have to use my Atari?" I muttered under my breath. I decided right then to write a program to print labels for my disks so that I could find my programs at all times.

Thus was born the following program, though it went through many revisions afterwards. This version takes advantage of some features of the Epson/Graftrax Plus chip set. Learning to use these features with the Atari is an important part of this story.

Graftrax Plus is an enhanced version of the graphics chips for the Epson MX series of printers. It replaces some less useful features of the previous version with some very useful ones. The TRS-80 graphics characters are gone, and added are true underlining, superscript and subscript characters. Inverse-video letters are printed as italics.

My big interest was in using the superscript/subscript font to pack as many filenames on a label as I could. By combining the very small superscript font and condensed print with the ability to reduce the spacing between lines, I reasoned I could make labels with lots of filenames and still have them be readable. I would also have a practical excuse for learning how to use these features of the Epson with my Atari.

Let's look at the superscript/subscript font. It produces the same number of characters per line as the regular font does, but uses only four of the nine pins in the printhead. It does not create superscript characters that have portions printed above the level of the rest of the text, but there is no doubt that it is indeed a superscript font. Here's an example of how it looks:

Figure 1

SUPERSCRIPT NORMAL

To achieve such a nice appearance, the printer prints the superscript characters in two passes, picking up different parts of the letters on each pass. You activate this feature by using the command PRINT #3;CHR$(27);CHR$(83);CHR$(0). It is strongly recommended that you use commands in the form PRINT #N;, rather than LPRINT, because the LPRINT command can pad lines with extra spaces and mess up positioning. The PRINT #N; is preceded, of course, by an OPEN #N,8,0,"P:" to open a channel to the printer; it is followed, at the end of the program, by a CLOSE #N. CHR$(27) is the ESCAPE code, which is used with many printer commands to warn the Epson that a control code is on the way. CHR$(83);CHR$(0) then puts the printer in the superscript mode; CHR$(83);CHR$(1) puts it into the subscript mode. If you need to turn off the superscript/subscript mode, just use PRINT #N;CHR$(27);CHR$(84).

Next, we have the problem of line spacing. This, it turns out, really involves two questions: how close together we want the lines, and what spacing can we use to ensure that we print at the same place on each successive label.

This second question is very important. Adhesive labels are manufactured with a typical spacing of one inch from the top of one label to the top of the next. We are going to pack as much print as we can onto each label, and it's very annoying to have to manually readjust the head position. We do have to readjust, however, if the printing has worked its way off the labels. The Epson has a mode that allows you to control the line spacing in increments of 1/216 of an inch (how's that for fine control!). After playing around with the superscript printing for a while, I determined that using 14/216-of-an-

inch line spacing would pack the absolute maximum of readable information on each label. My first attempt to use this spacing, however, revealed the need to consider label-to-label spacing.

The problem comes down to this: we want to pack as much as we can onto a label (so spacing can't be less than 14/216) and we don't *want to* go to more than 36/216 (or 1/6, which is the "normal" line spacing). Also, whatever integer we choose must divide evenly into 216 so that the number of lines in one inch will be an integral number.

A little simple math shows that the only values we can use are 18, 24 and 36, and I used 18 since it is the smallest of the possible integers. This gives us twelve lines from the top of one label to the top of the next (eleven on each label and one between labels). If you want one-dot spacing between lines, as in Figure 2, replace the 18 in line 33 with 24. I don't print on the top of each label, though, because it's too much of a bother to line up the printhead and the labels with the accuracy needed to use this space.

Each label has a title line, which leaves nine lines for filenames. I use 5″ labels, because they fit nicely on disk envelopes and permit me to print names four across (in the condensed-print mode), for a total of 36 per label. Figure 2 is an example of this:

Figure 2

```
DISK = DISK 1
* VULTURES     081  * CASTLE  HEX 087  * DOS     SYS 039  * ADVENTURFNT 009
* INIT         003  * ADVENTUR    098  * WEST1       003  * NORTH       004
* NORTH1       004  * CAVE5       002  * CAVE2       002  * CAVE3       003
* CAVE4        003  * SOUTH2      004  * SOUTH1      002  * TREE        001
* SOUTH4       004  * SOUTH3      002  * WEST3       003  * WEST2       003
* CAVE6        002  * CAVE7       004  * DEFAULT     002  * CAVE8       003
* CAVE10       003  * MONSTER     002  * CAVE11      002  * CAVE11A     003
* CAVE9        005  * CAVE12      002  * PASS1       002  * PASS2       002
* CAVE15       003  * CAVE16      002  * PASS3       003  * CAVE14      003
```

My program also moves to the next label if you have more than 36 files on a single disk, and continues to print. Since this second label needs no title, there is room for 40 more filenames on it.

If you don't like to stick labels on your disk envelopes (they are, after all, not that easy to peel off), you can use this program to print on plain paper, and then just tape the printout paper onto the envelope or disk. In this case, just change line 137 to read:

```
IF RECCTR=40
THEN RECCTR=0
```

If you don't have access to 5″ labels, and can only get the more common 3½″ variety, make the following changes in the program: Change 4 to 3 in lines 60, 135 and 312; change 40 to 32 in line 137. Also, change line 130 so that there is only one space between the first set of quotes. You will then get 27 names on the first label and 30 names on the second.

The program must keep track of where you are in the count (of labels and lines) to get properly positioned for the next label. It must count how many filenames it is sending for each line of print and how many lines it has printed. The variables CTR and RECCTR keep track of these values. They are reset to zero at the end of each line, and the end of each label respectively.

The program looks for the FREE SECTOR file, the last one on each disk, as an end signal. B$ is the string that contains the filenames as they are read off the disk directory. B$(4,5) will be " F" if, and only if, the FREE SECTOR file has been reached. At that point, the program goes on to determine how many blank lines it needs to print to get to the next label. It does this by using MARKER, which tells it how many lines have been printed.

I have used this program extensively to print labels for my disk envelopes and it has made finding programs much easier. An enterprising programmer might want to add a routine to alphabetize the names on each label, since there are still precious moments wasted in scanning a label with 30 or 40 names on it to find the particular filename of interest.

```
1 REM SUPER DISK LABEL PROGRAM by Richard Kushner
2 REM ANTIC MAGAZINE, JANUARY 1984
10 DIM B$(17),N$(40)
20 ? "⇱":? "LOAD PRINTER WITH LABELS.":?
21 ? "TURN ON PRINTER/850 INTERFACE UNIT.":?
22 ? "INSERT DISKETTE INTO DRIVE #1.":?
23 ? "PRESS RETURN WHEN READY.":?
25 INPUT N$
28 TRAP 500
30 OPEN #7,8,0,"P:"
33 PUT #7,27:PUT #7,51:PUT #7,18
37 ? #7;CHR$(15);:? #7;CHR$(27);CHR$(83);
40 CLOSE #2:? "ENTER DISKETTE NAME (RETURN FOR END)":INPUT N$
42 IF N$="" THEN CLOSE #2:CLOSE #7:END
45 X=0:N=0:CTR=0:RECCTR=0:INDEX=13
50 ? #7;" DISK = ";N$
60 RECCTR=RECCTR+4
100 OPEN #2,6,0,"D:*.*"
120 TRAP 310:INPUT #2,B$
130 ? #7;B$;"  ";:CTR=CTR+1:RECCTR=RECCTR+1:IF B$(4,5)=" F" THEN 310
135 IF CTR=4 THEN PRINT #7:CTR=0
137 IF RECCTR=40 THEN PRINT #7:PRINT #7:RECCTR=0
140 GOTO 120
310 IF RECCTR=0 THEN GOTO 320
312 MARKER=INT(RECCTR/4)
314 IF CTR<>4 THEN MARKER=MARKER+1
316 FOR I=1 TO INDEX-MARKER:PRINT #7:NEXT I
320 GOTO 40
500 TRAP 40000:IF PEEK(195)=138 THEN 20
510 ? "ERROR #";PEEK(195);" AT LINE ";PEEK(186)+256*PEEK(187):STOP
```

TYPO TABLE

Variable checksum = 182315

Line num range	Code	Length
1 – 37	AQ	449
40 – 310	YC	474
312 – 510	YX	209

GIVE YOUR KIDS A LESSON THEY'LL NEVER FORGET.



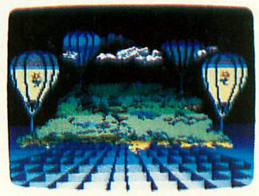
When kids have fun and learn at the same time, they're more likely to remember more of what they've learned. What's more, when they associate the two together — learning becomes an enjoyable activity. So they'll do more of it.

That's the basic principle behind the Learning Fun Series from EPYX.

FUN WITH MUSIC. PARENTS TAKE NOTE.

Fun with Music™ is designed to appeal to both you and your children. It comes with a songbook and has two modes of play. In the first mode, you can enter songs from the songbook — or compose your own. Then play the songs back, adding or removing notes and changing tempo or key. Your computer is like a musical instrument with memory, and you see every note of it displayed on your screen.

The second mode lets you play your song in a fun-filled action game. You control a drum major trying to touch the notes before a small but pesky poodle catches up to him and slows down the parade.





Either way, Fun with Music gives you and your whole family the perfect mix of learning and play.

FUN WITH ART. JUST PICTURE IT.

Creating art on the video screen is one of the newest forms of "high-tech" play for kids. And adults, for that matter. Plus there's nothing to clean up afterwards.

Fun with Art™ uses the computer and joystick to transform your TV screen into an artist's canvas with astonishing results. 128 colors, numerous brush strokes, all kinds of geometric shapes, and special fill-in and magnifying options are some of the 24 different modes and features available to create works of art never before possible.

Fun with Art brings out the artist in you, no matter what your age!

MORE LEARNING FUN ON THE WAY.

These two are the first of an extensive series of Learning Fun games we have planned. Look for these, as well as other EPYX titles, wherever computer software is sold.

EPYX™ COMPUTER SOFTWARE

LEARNING FUN SERIES.

ROM FUN

Survey of recent cartridge games

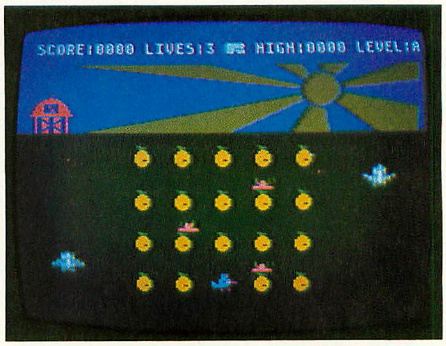
by DAVID DUBERMAN

We've been receiving a steady stream of game software in the form of ROM cartridges lately. Games on cartridge offer several advantages to the Atari owner, including speed of loading, lower memory requirements, and convenience. Since most Atari 400's are equipped with at least 16K of RAM, all cartridges run on the bottom-of-the-line computer. Here are some brief descriptions of the more noteworthy games we've received.

ROMOX, INC.
476 Vandell Way
Campbell, CA 95008
(408) 374-7200
8K Cartridge
$44.95

Flapper is Marc Russel Benioff's first attempt at writing an arcade game, and not a bad one at that. Your player is a bird in a garden, and your object is to maneuver close enough to each of the 24 fruits in the garden so that your bird can pop them with its beak. When you've done this, you are challenged with a new array of fruit. Meanwhile, two farmers cruise up and down the sides of the garden, throwing rocks at you. You are also fighting time — the sun literally flies through the sky — and when it sets, you lose a life and must start with a fresh, full garden.

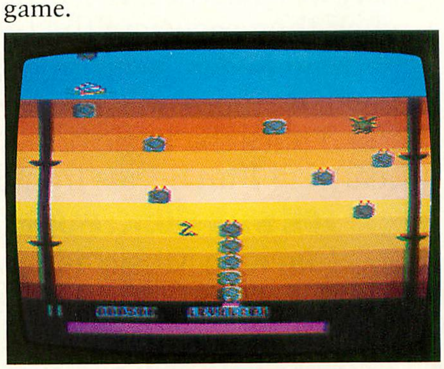
Player movement is a novelty — push the stick to either side for lateral movement, and push the fire button to move upwards, a flap at a time. The only way to move downwards is to await gravity's



Flapper

inevitable tug. You must plan your movements carefully to avoid finding yourself between a rock and a hard place (so to speak). Accompanied throughout by a jaunty tune penned by Marc's grandmother, Flapper is fun, and a relatively non-violent arcade entertainment.

Also from Romox comes **Spark Bugs**, featuring a unique play mechanic involving various insects and electricity. Spark Bugs is a fast-paced, lively and colorful game.



Spark Bugs

ACTIVISION

2350 Bay Shore Frontage Rd.
Mountain View, CA 94043
(415) 960-0410
16K Cartridge
$34.95

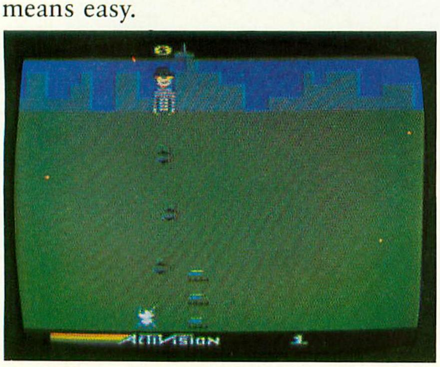
River Raid is Carol Shaw's adaptation of her hit game for Atari VCS. This is your basic, vertically-scrolling shoot-em-up, with a lot of graphics frills thrown in. Your object is primarily to survive as long as possible, while shooting every-



River Raid

thing in sight. River Raid is difficult at the start, and becomes increasingly challenging as the game progresses. Shaw has spiffed up the game visuals somewhat for the computer, but the concept remains identical to that of the VCS original.

Also from Activision, **Kaboom!**, another VCS conversion, involves catching falling bombs with laterally moving paddles. The graphics are cute, and the play action is simple, but not by any means easy.



Kaboom!

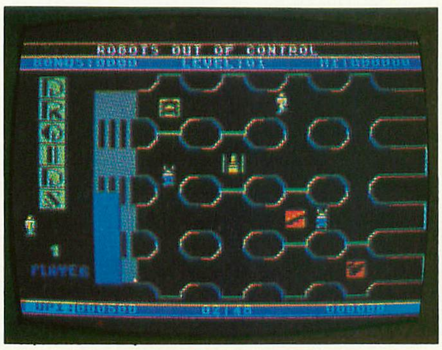
TG PRODUCTS

1104 Summit Ave., Suite 110
Plano, TX 75074
(214) 424-8568
16K Cartridge
$34.95

In **Droids**, you find yourself in command of a futuristic space transport which has just undergone the rigors of an ion storm. Your crew of normally helpful droids has gone beserk, and is attempting to disable your ship by deactivating the various support systems. Armed only with an integrator-beam gun, your task is to return your ship's systems to the proper working order.

The ship's layout is maze-like, with droid compartments at one end, and your transfer shaft at the other. At the base of the shaft is the device to which you must return for energy every time you reactivate a system. The four systems are engineering, weapons control, life support, and communications. When you shoot a droid, it returns to is compartment briefly, then goes berserk again.

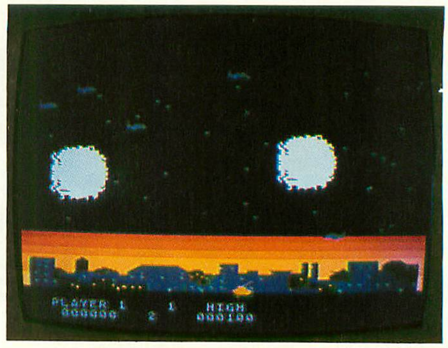
When a system is deactivated, it turns red. To reactivate it, you must pass over it twice, each time with a new charge of



Droids

energy, without a droid having touched it. When you have fully reactivated all the systems you pass to the next level, where your task is similar, but more difficult.

In **Night Strike**, you are entrusted with the defense of your city against a vicious enemy. Armed with a mobile anti-aircraft tank and different types of weaponry, you must rid the skies of enemy aircraft. Flak shells, flares, and 20mm anti-aircraft rounds are at your



Night Strike

disposal. Night Strike adds several new wrinkles to the Missile Command theme, resulting in a fine challenge.

Both Droids and Night Strike were written by The Solitaire Group, an organization of programmers.

IMAGIC

981 University Avenue
Los Gatos, CA 95030
(408) 399-2200
8K Cartridge
$29.00

From Imagic comes **Atlantis** by Dave Johnson, also a conversion from a VCS game. You command three defense posts that guard the underwater city against aerial invasions. The dreaded Gorgon Fleet is attacking, and you must destroy each ship before it reaches the level from which it can vaporize one of Atlantis' buildings. The play mechanic is unusual



Atlantis

— you fire straight up with a gun in the center of the screen, and at a 30-degree angle towards each opposite upper corner from two guns situated at the centers of the screen's right and left edges. You can't change the guns' angles or positions; you can only determine which gun will fire when you press the button by moving (or centering) the joystick. Graphics are fairly sparse but very attractive and colorful, but the play mechanic may take a bit of getting used to.

Turn your Atari into a Ferrari.

Introducing the all-new 1984 Indus GT™ disk drive. The most advanced, most complete, most handsome disk drive in the world.

A flick of its "Power" switch can turn your Atari into a Ferrari.

Looks like a Ferrari.
The Indus GT is only 2.65" high. But under its front-loading front end is slimline engineering with a distinctive European-Gran flair.

Touch its LED-lit CommandPost™ function control AccuTouch™ buttons. Marvel at how responsive it makes every Atari home computer.

Drives like a Rolls.
Nestled into its soundproofed chassis is the quietest and most powerful disk drive power system money can buy. At top speed, it's virtually unhearable. Whisper quiet.

Flat out, the GT will drive your Atari track-to-track 0-39 in less than one second. And when you shift into SynchroMesh DataTransfer,™ you'll increase your Atari's baud rate an incredible 400%. (Faster than any other Atari system drive.)

And, included as standard equipment, each comes with the exclusive GT DrivingSystem™ of software programs. World-class word processing is a breeze with the GT Estate WordProcessor.™ And your dealer will describe the two additional programs that allow GT owners to accelerate their computer driving skills.

Also, the 1984 Indus GT is covered with the GT PortaCase.™ A stylish case that conveniently doubles as a 80-disk storage file.

Parks like a Beetle.
The GT's small, sleek, condensed size makes it easy to park.

And its low $449 price makes it easy to buy.

So see and test drive the incredible new 1984 Indus GT at your nearest computer dealer soon.

The drive will be well worth it.



INDUS™

The all-new 1984 Indus GT Disk Drive.
The most advanced, most handsome disk drive in the world.



For additional information, call 1-800-33-INDUS. In California, call 1-800-54-INDUS. 213/882-9600.

ATARI AND EPSON

A matter of control

by DOUGLAS MACKAY

The popular Epson MX series of printers is widely used by Atari owners. The Epson printers are reliable, versatile, and if equipped with the "graftrax" option, their superb graphics complement any Atari computer.

Still, many of us users have trouble when we try to get our Epsons to respond to the script commands of AtariWriter or the Atari Word Processor. The latter program has five script commands, or *fonts* as they are properly called, and the printer has 22 script modes, plus a variety of other features. Unfortunately, when combined these products fail to communicate fully with each other. This is very frustrating.

The Epson manual helps a bit. It explains that any print command can be sent to the printer prior to loading a word processor. That command remains "on" until the printer is reset. The procedure is to load BASIC, send the specific code to the printer for the desired script, remove BASIC and then boot the word processor. This is a nuisance because the word processor has to be dumped, and the whole procedure performed again, to set up another typing mode.

However, owners of Atari Word Processors and Epson printers need not despair – the problem can be overcome. The solution is not 100% effective, but it is easy, and will allow you to use about 95 percent of your printer's capability.

All of the Epson's capabilities are activated by sending a specific control code from the computer to the printer. The codes simply turn particular functions on and off. You can find these codes in the appendix to the Epson manual. The problem is that the Atari Word Processor has been programmed to assume that the printer is a Centronics-type printer. (The Atari 825 printer was made by Centronics.) The codes that activate it are different from the Epson code set.

Nevertheless, the Atari Word Processor is capable of sending control codes other than those imbedded in its program: codes to which the Epson will respond. The procedure is begun by pressing the [CTRL] and [INSERT] keys simultaneously, which tells the system that a control character is about to be entered. Using the chart at the end of this article, or the Epson manual, you then enter the code for the desired characteristic. The codes are alphanumeric symbols usually preceded by the [ESC] key. Remember all alphabetic codes must be in capitals. If one characteristic is to be combined with another, simply repeat the procedure – [CTRL] [INSERT] followed by the second code. If at some point in your text you wish to change modes, merely set up the corresponding "off" code in exactly the same manner. If another script or mode is desired, follow the "off" command with the new "on" command. It's really very easy.

The codes can be entered at any location in your word processed text. It is suggested that you edit, format and save your document first. Then just prior to printing, add the printer control codes. The codes will appear on the screen but will not be printed. They will also seem to upset your format by moving the following text two spaces to the right. Again, this will not be apparent on the printout. If you wish to preserve the control codes in your document, save each page again to disk.

There are two unsolved problems that remain with the Atari Word Processor/Epson combination. One, *subscript* is not available. Epson's control format requires that the *superscript* code be followed by a 0 to create the superscript mode. The word processor's "Special Character Insert" does not allow two alphanumeric symbols to be entered at one time. The second problem is with *underlining*. It is easy to send the code to activate underlining; the problem is turning it off. Again,

continued on next page

Table 1

REFERENCE CHART FOR EPSON MX-80 AND ATARI WORD PROCESSOR

Compressed Mode:			
ON —	[CTRL] [INSERT]	[CTRL] [INSERT]	
OFF —	[CTRL] [INSERT]	[CTRL]	[R]
Double Strike Mode:			
ON —	[CTRL] [INSERT]	[ESC]	[G]
OFF —	[CTRL] [INSERT]	[ESC]	[H]
Double Width Mode: (will only stay on for maximum one line at a time)			
ON —	[CTRL] [INSERT]	[TAB]	
OFF —	[CTRL] INSERT]	[DELETE/BACK SPACE]	
Emphasized Mode:			
ON —	[CTRL] [INSERT]	[ESC]	[E]
OFF —	[CTRL] [INSERT]	[ESC]	[F]
Italics Mode: (not underline)			
ON —	[CTRL] [INSERT]	[ESC]	[4]
OFF —	[CTRL] [INSERT]	[ESC]	[5]
Subscript Mode:			
ON —	[CTRL] [INSERT]	[ESC]	[S]
OFF —	[CTRL] [INSERT]	[ESC]	[H] OR [T]
Superscript Mode:			
	UNAVAILABLE		
Underline Mode:			
ON —	[CTRL] [INSERT]	[ESC]	[—]
OFF —	[CTRL] [INSERT]	[ESC]	[@]
SKIP BLANK SPACES	[CTRL] [INSERT]	[ESC]	[F]
Line Spacing 1/6": (Default Mode)			
ON —	[CTRL] [INSERT]	[ESC]	[2]
Line Spacing 1/8":			
ON —	[CTRL] [INSERT]	[ESC]	[0]
OFF —	[CTRL] [INSERT]	[ESC]	[2]
Line Spacing 7/72":			
ON —	[CTRL] [INSERT]	[ESC]	[1]
OFF —	[CTRL] [INSERT]	[ESC]	[2]
System Reset and New Top of Form:			
	[CTRL] [INSERT]	[ESC]	[@]
Top of Form:			
	[CTRL] [INSERT]	[CTRL]	[L]

Table 2

REFERENCE CHART FOR EPSON MX-80 AND ATARIWRITER

Compressed Mode:							
ON —	[CTRL] [O]	15					
OFF —	[CTRL] [O]	18					
Double Strike Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	71			
OFF —	[CTRL] [O]	27	[CTRL] [O]	72			
Double Width Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	87	[CTRL] [O]	1	
OFF —	[CTRL] [O]	27	[CTRL] [O]	87	[CTRL] [O]	0	
Emphasized Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	69			
OFF —	[CTRL] [O]	27	[CTRL] [O]	70			
Italics Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	52			
OFF —	[CTRL] [O]	27	[CTRL] [O]	53			
Subscript Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	84	[CTRL] [O]	1	
OFF —	[CTRL] [O]	27	[CTRL] [O]	72			
Superscript Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	84	[CTRL] [O]	0	
OFF —	[CTRL] [O]	27	[CTRL] [O]	72			
Underline Mode:							
ON —	[CTRL] [O]	27	[CTRL] [O]	45			
OFF —	[CTRL] [O]	27	[CTRL] [O]	45	[CTRL] [O]	0	
Line Spacing 1/6": (Default Mode)							
ON —	[CTRL] [O]	27	[CTRL] [O]	50			
Line Spacing 1/8":							
ON —	[CTRL] [O]	27	[CTRL] [O]	48			
OFF —	[CTRL] [O]	27	[CTRL] [O]	50			
Line Spacing 7/72":							
ON —	[CTRL] [O]	27	[CTRL] [O]	49			
OFF —	[CTRL] [O]	27	[CTRL] [O]	50			
System Reset and New Top of Form:							
	[CTRL] [O]	27	[CTRL] [O]	64			

Epson's control format requires the underline "on" command to be followed by 0 to turn it off. There is an exception that is occasionally useful. Control code [ESC] [@] is a system reset and can be used as an extreme method to turn off the underlining mode. It is extreme because when the printer receives [ESC] [@], it immediately resets all modes to default, jumps down one line and resumes printing from the left-hand side. Any text on the line that came after the [ESC] [@] is lost. However, you will find it works well when headings are underlined, or in any other situation where the printer has room to reset before it resumes printing.

ATARIWRITER

The new AtariWriter was designed to be compatible with all four Atari printers and most of the popular printers on the market. The manual suggests that non-Atari printer owners obtain a "printer-driver" program from Atari Program Exchange. The good news is that Epson MX owners do not need a printer-driver program because it is possible to access every feature of your printer directly from AtariWriter.

When prompted by AtariWriter as to which Atari printer you are using, respond with #3, "Atari 820." If you do noth-

ing more than this, you will get 80 columns in normal script on your Epson. However, the Epson will still not respond directly to the print commands from the AtariWriter program. To access the multiple capabilities of the printer, it is necessary to send control codes by alternate means. This is not a problem; AtariWriter makes it easy to enter and send Epson control codes.

Again, first edit, then format and save your letter or document. Prior to printing, add the required print codes at any desired place in the text. In the case of AtariWriter, all commands must be sent in their decimal equivalent. These codes are listed in the appendix to the Epson manual, and I have included a list of the more frequently-used codes at the end of this article.

The procedure is to press [CTRL] and [O] simultaneously, which sets the system up to receive a control code. Since most Epson codes must be prefaced by [ESC] and the decimal equivalent of [ESC] is 27, you will enter: [CTRL] [O] 27 and then [CTRL] [O] followed by the specific decimal command code. If several print modes are to be combined, enter the whole first code followed by the whole second code. It may seem that the codes are particularly long, but in fact they are easy to enter, and a lot cheaper than buying a printer-driver program. Because of this code format, every feature of the Epson can be utilized without problem, including subscript and underlining. As with the Atari Word Processor, the control codes that appear on the screen will not be printed and will not upset your text format.

Supermarket shoppers are already familiar with "word-find" puzzle books, which grace the checkout-line magazine racks of America.

The program presented here will enable you to make your own custom word puzzle on any Atari equipped with a disk or cassette, and a printer (40- or 80- column). This puzzle might prove useful as a spelling study aid, as a quiet alternative to video games, or as party entertainment. In fact, I originally wrote this program for a friend who needed it for a sales promotion party.

Before typing in the program listing for Word Find, note the changes required for your printer. The "PW" equate in line 60 indicates the printer width for centering the printouts. If you have a 40- or 80-column printer, change this line accordingly. A 40-column printer will require the following code changes as well:

```
1150  C=C+1:IF C=2 THEN C=0:? #3
1460  $ #3;PUZZLE$(A,A);
```

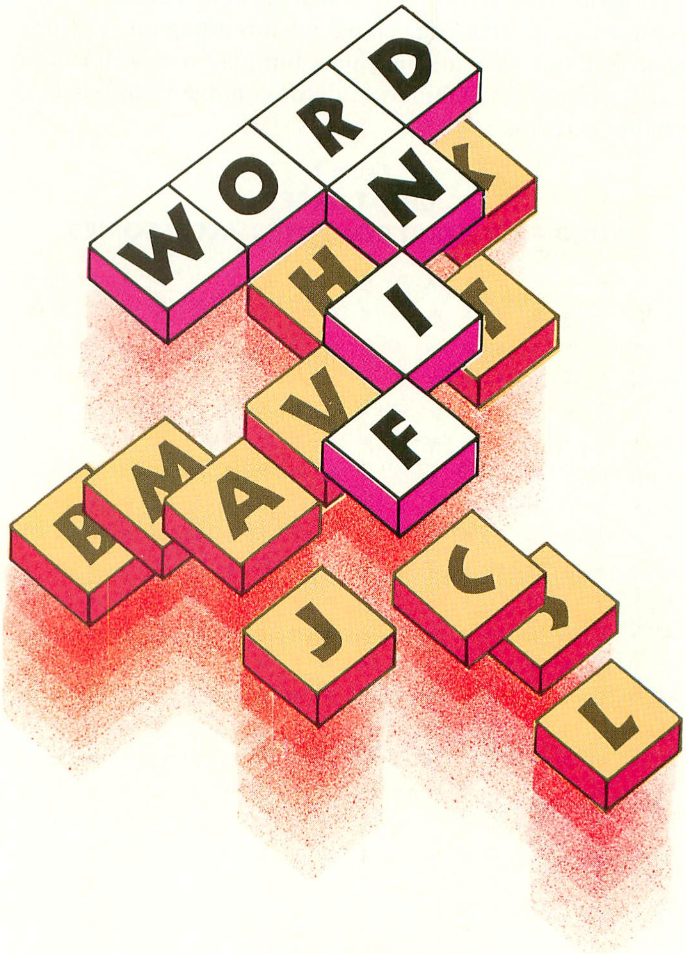
This will put two words per line in the word list, and will change the puzzle from double to single spacing horizontally. Double spacing is used to help make the Word Find puzzle easier to read.

When the program is run, you will be asked for a puzzle size between 10 and 30, then you will be requested to input the filename that your word list is on. Type NEW to go to the subroutine for creating a new word file. This can be put to cassette, C:, or a disk file, D:name. Type END when you are through inputting new words. You must then input the title of the Word Find puzzle that will be created, and select difficult or easy puzzles. The program will place words in the letter matrix vertically, horizontally and diagonally for "easy" puzzles and at any angle, with 45-degree increments, between zero and 360 degrees for "difficult" puzzles.

You will then be asked for the number of copies and, if more than one is desired, whether different or identical puzzles should be printed. Different puzzles will use the same word list, but the words will be hidden differently. Finally, you may select cheat sheets if you wish, an example of which is shown in Figure 1. This is a printout of the puzzle after all the words have been placed, but before the asterisks are replaced with random letters. Figure 2 is an example of a completed Word Find puzzle.

Sometimes a word read from the input file will not fit in the puzzle string, PUZZLE$. The variable TEST is used to keep track of how many times the program selects a vector and beginning position, and attempts to fit WORD$ into PUZZLE$. The program will try more times as the word count increases, because it will be increasingly difficult to find an open spot in the letter matrix as each word is placed. Once the program has tried the maximum number of allowed attempts without success, the word will be thrown out and deleted from the printed word list in the final Word Find puzzle. Generally, if the puzzle size is equal to the total number of characters in the longest word to be placed, most, if not all, of the words will fit. Making the puzzle oversized will probably insure that all the words can be placed, but may increase the difficulty of word finding.

The program limits the number of words in any puzzle to



by MATTHEW RATCLIFF

three times the puzzle size. Maximum word length is limited to 14 characters (see line 1670), with the fifteenth character position in WORD$ always blank to format the output. This can be expanded up to 20 characters for 80-column printers by modifying the DIM's for WORD$ and WSV$, and by changing the 15's in lines 570, 600, and 1140 to the new maximum word size. Line 1670 must be changed as well.

When creating a word file, all printable characters (on your printer) are allowed, including punctuation, except for the asterisk (*) and comma (,). The asterisk is used to indicate "this space is empty" in PUZZLE$, when filling it with the words read from the input file. The comma will print out to the word file properly, but will input from the file as separate words, dropping the comma altogether (not as a phrase with an embedded comma, as intended). Lower case characters are also allowed, if your printer can handle them. Note that the LTR$ equate in line 70 only includes upper case characters A through Z. Should any different characters be put in the word file, such as lower case or spaces, they should be added to this equate. The random-letter fill routine will then include these characters from LTR$ in the puzzle matrix as well, so that your "special characters" will not make the locations of these words obvious.

continued on next page

Once you have this program typed in and checked out with TYPO, change the printer setup code and feel free to customize Word Find in any way you wish. The code is heavily commented, so that you can tell what is going on. The program will run on a 16K machine, but may very well run on an 8K machine by eliminating all the comments and making smaller puzzles.

Figure 1

CHEAT SHEET
WORD FIND ATARI COMPUTER ACCESSORIES

```
* * * * * * * M A R G O R P * * * * * *
* C * * * * * * * * * * S E L B A C * *
I * I * * * * * * * * P M X * C * * * *
* R * T * * * * * * R A * I Y I * * * J
* * A * S * * * * I G * K R R R * * * O
* * * T * U * * N * * * E T O E * * * Y
* R E P A P O T * D E * Y A M M * * * S
* * * * * * E C * * O C P M E U * * * T
* * * * * R * * A * * T A * M N * * * I
* * * * S * Y T R U O F D F * * * * * C
* E V I R D * * C * * * * * R R * * * K
* * C * * * * * O * * * * * * E * * * S
* * O * * * * * M * * L * * * C T * * *
M * L * * * * * P P * * A * * O * N * *
E * U * * * * * U A * * * M * R * * I *
D * M * K S I D T D M A R * R D * * * *
O * N * * * * * E D * * O * * E * * * *
M * * * * * * * R L * * M * * R H * * *
* * * * * * * * S E * * * * * * * T * *
* * * * * * * * * S M O D U L E * * * *
```

Figure 2

COMPLETED PUZZLE
WORD FIND ATARI COMPUTER ACCESSORIES

```
U L J V K X X M A R G O R P P B A G A B
O C E U H D D Z N S S T S E L B A C T C
I H I V O N L C C P O P M X O C L O B S
I R C T H D K K O D R A Z I Y I K P N J
X Z A F S F F G N I G A K R R R Y F X O
P D R T D U Z P N R X N E T O E Y C D Y
Z R E P A P O T B D E B Y A M M B L L S
S K I Y Y X E C K J O C P M E U Y P D T
I D U I M R X N A W U T A Z M N E T U I
H P T E S J Y T R U O F D F N Q D M B C
R E V I R D F N C V C B Z G R R L N M K
W C C W W R P N O L A Y M L B E W B U S
Q I O S C L H N M V N L C P A C T B N L
M H L L H V N A P P O E A X S O F N C I
E S U L G E A B U A B T B M B R C V I U
D B M W K S I D T D M A R H R D Z T B X
O M N R J Y K P E D G Z O O L E Z X O I
M U D J U W E F R L B O M C D R H N X S
M K O E H F K C S E N C L T R Q B T Q X
C F W M F M S C R S M O D U L E N L Q G
```

ATARI	COMPUTERS	PROGRAM	RECORDER
DISK	DRIVE	PRINTERS	FOURTY
COLUMN	THERMAL	DOT	MATRIX
ACOUSTIC	MODEM	INTERFACE	MODULE
MEMORY	JOYSTICKS	PADDLES	PAPER
CABLES	NUMERIC	KEYPAD	GAME
RAM	ROM		

REQUIRES 16K RAM

```
10 REM ATARI WORD FIND PUZZLE GENERATO
R
11 REM BY MATTHEW RATCLIFF
12 REM ANTIC MAGAZINE
13 REM JANUARY, 1984
20 GRAPHICS 0:? "* WORD FIND BY MAT RA
T *":TRAP 1800
30 ? :? "PUZZLE SIZE (10-30) ";:INPUT
PSZ
40 IF (PSZ<10 OR PSZ>30) THEN ? PSZ;"
SIZE ERROR !":GOTO 30
50 LPZ=PSZ*PSZ:DIM PUZZLE$(LPZ),WORD$(
15),LTR$(80),FILE$(15)
60 DIM Q$(5),WSV$(3*PSZ*15),TTL$(50):P
W=72:REM PRINTER WIDTH
70 LTR$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
80 LLTR=LEN(LTR$):REM YOU CAN ADD SPEC
IAL CHARACTERS TO
90 REM THE RANDOM LETTER FILL ARRAY, L
TR$ TO ALLOW FOR
100 REM HYPHENATED WORDS & SHORT PHRAS
ES IN THE LIST. ONLY
110 REM THE '*' & "," ARE NOT ALLOWED
 IN LTR$
120 WCNT=0:REM MAX WORDS ALLOWED, 3*PS
Z
130 REM
140 ? :? "INPUT WORD FILENAME,"
150 ? "(C: FOR CASSETTE), 'NEW' TO CRE
ATE"
160 ? "A NEW WORD FILE ";:INPUT FILE$
170 IF LEN(FILE$)=0 THEN ? :? "NO FILE
 NAME ?  NO GOOD -":GOTO 140
180 IF FILE$="NEW" THEN GOSUB 1530
190 FLG=1:IF FILE$(1,1)="C" THEN FLG=0
:REM CASSETTE FLAG
200 IF FLG THEN 240
210 ? :? "REWIND CASSETTE WITH WORD FI
LE ON"
220 ? "IT. PRESS PLAY AND PRESS RETURN
 KEY"
230 ? "WHEN CONSOLE SPEAKER BEEPS ONCE
."
240 OPEN #5,4,0,FILE$
250 IF FLG THEN 280
260 FOR I=1 TO 128:GET #5,A:NEXT I
270 REM GET DUMMY RECORD TO STOP CASSE
TTE
280 ? :? "INPUT TITLE OF PUZZLE ";:INP
UT TTL$
290 ? :? "(D)IFFICULT OR (E)ASY PUZZLE
S ";:INPUT Q$:IF LEN(Q$)=0 THEN 290
300 DIF=2*(Q$(1,1)="D")+(Q$(1,1)="E")
310 IF DIF=0 THEN ? Q$;" * WHAT ? *":G
OTO 290
320 ? :? "INPUT NUMBER OF PUZZLE COPIE
```

```
S ";:INPUT NCOP
330 IF NCOP<1 THEN ? "HOW MANY ?":GOTO
 320
340 IF NCOP=1 THEN 400
350 ? :? "(I)DENTICAL OR (D)IFFERENT C
OPIES ";:INPUT Q$
360 TYP=2*(Q$(1,1)="I")+(Q$(1,1)="D")
370 IF TYP=0 THEN ? "'I' OR 'D', NOT "
;Q$:GOTO 350
380 REM PRINTER SET-UP CODE.  REPLACE
WITH YOUR
390 REM PRINTER'S SPECIFIC REQUIREMENT
S.
400 OPEN #3,8,0,"P:"
430 REM * ASK FOR CHEAT SHEETS *
440 ? :? "WOULD YOU LIKE CHEAT SHEETS
(EXTRA"
450 ? "PRINTOUT OF PUZZLE BEFORE FILLI
NG"
460 ? "WITH RANDOM LETTERS), Y/N";:INP
UT Q$:IF LEN(Q$)=0 THEN 440
470 CHEAT=2*(Q$(1,1)="N")+(Q$(1,1)="Y"
)
480 IF CHEAT=0 THEN ? "'Y' OR 'N' ?":G
OTO 440
490 REM * FILL PUZZLE WITH '*'S FIRST
 *
500 PUZZLE$(1,1)="*":PUZZLE$(LPZ,LPZ)=
"*":PUZZLE$(2)=PUZZLE$
510 REM * GET A WORD AND CHECK FOR 'EN
D' *
520 INPUT #5,WORD$:? WORD$
530 IF WORD$="END" THEN GOTO 1010
540 LW=LEN(WORD$)
550 REM *PAD WITH BLANKS TO 15 CHARACT
ERS *
560 REM * TO FORMAT OUTPUT,& SAVE IN W
SV$ *
570 FOR I=LW+1 TO 15:WORD$(I,I)=" ":NE
XT I
580 WCNT=WCNT+1:REM WORD COUNT
590 IF WCNT>3*PSZ THEN ? "* ERROR *":?
 "WORD FILE TOO LONG...":END
600 WSV$(15*(WCNT-1)+1,15*(WCNT-1)+15)
=WORD$
610 REM
620 REM * PICK RANDOM COORDINATES AND
VECTOR *
630 REM * VECTOR IS 0,+45, OR -45 DEGR
EES      *
640 REM * FOR EASY PUZZLES
         *
650 TEST=0:REM TEST COUNTER FOR # OF T
IMES WE
660 REM TRY TO FIT WORD$ IN PUZZLE$
670 X=INT(PSZ*RND(0))+1:Y=INT(PSZ*RND(
0))+1
680 TEST=TEST+1
690 IF TEST<(WCNT+1)*5+10 THEN GOTO 74
0
700 REM TRY MORE TIMES WHEN PUZZLE IS
NEARLY FULL
710 WCNT=WCNT-1:IF WCNT<1 THEN WCNT=1:
REM THROW IT OUT
720 ? :? WORD$;" * WONT FIT *"
730 GOTO 520
740 ON DIF GOTO 750,760
750 V=INT(RND(0)*3)+2:GOTO 790
760 V=INT(RND(0)*8)+1
770 REM SET X1,Y1 - X,Y INCREMENTS FOR
780 REM CHOSEN DIRECTION
790 X1=(V=2 OR V=3 OR V=4)-(V=6 OR V=7
 OR V=8)
800 Y1=(V=1 OR V=2 OR V=8)-(V=4 OR V=5
 OR V=6)
810 REM * SEE IF WORD WILL FIT THERE *
820 FOR I=1 TO LW
830 X2=X+I*X1
840 Y2=Y+I*Y1
850 IF (X2<1 OR Y2<1) THEN GOTO 670
860 IF (X2>PSZ OR Y2>PSZ) THEN GOTO 67
0
870 A=PSZ*(Y2-1)+X2
880 IF PUZZLE$(A,A)="*" THEN GOTO 910
890 IF PUZZLE$(A,A)=WORD$(I,I) THEN GO
TO 910
900 GOTO 670:REM SPACE ALREADY TAKEN,
TRY AGAIN
910 NEXT I
920 REM * OK, INSERT WORD HERE *
930 FOR I=1 TO LW
940 X2=X+X1*I:Y2=Y+Y1*I:A=PSZ*(Y2-1)+X
2
950 PUZZLE$(A,A)=WORD$(I,I)
960 NEXT I
970 GOTO 520:REM GET NEXT WORD UNTIL
END
980 REM
990 REM
1000 REM END OF FILE REACHED
1010 ON CHEAT GOTO 1020,1060
1020 GOSUB 1340:REM PRINT PUZZLE WITHO
UT RANDOM LETTER FILL
1030 REM
1040 REM NOW FILL PUZZLE$ WITH RANDOM
LETTERS
1050 REM
1060 FOR I=1 TO LPZ
1070 IF PUZZLE$(I,I)<>"*" THEN GOTO 11
00
1080 I2=INT(RND(0)*LLTR)+1:REM SELECT
A RANDOM CHARACTER
1090 PUZZLE$(I,I)=LTR$(I2,I2):REM FROM
 LTR$
1100 NEXT I
1110 REM * NOW PRINT OUT PUZZLE AND WO
RD LIST *
1120 GOSUB 1340:C=0
1130 FOR I=1 TO WCNT
1140 ? #3;WSV$(15*(I-1)+1,15*(I-1)+15)
;
1150 C=C+1:IF C=4 THEN C=0:? #3
1160 NEXT I:? #3:? #3:? #3:? #3
1170 REM * PRINT MULTIPLE IDENTICAL CO
PIES ?        *
1180 IF NCOP>1 AND TYP=2 THEN NCOP=NCO
P-1:GOTO 1120
```

continued on page 117

Six games that'll make your ATARI® the apple of your eye.

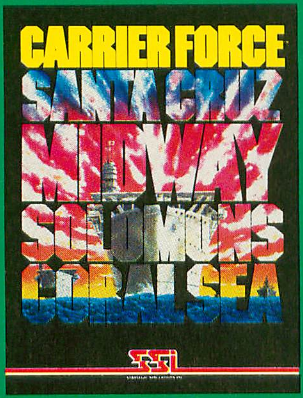
Playing games is probably one of the main reasons you bought an ATARI® home computer. When it comes to games, there are none more sophisticated, thought-provoking, challenging — and just plain fun — than the strategy games from SSI.

We made our considerable reputation by producing some of the finest games for the Apple®. To make sure all you ATARI owners out there don't get left out, we're converting as many of our games to your favorite computer as we possibly can.

Here are just six of our ever-increasing line of ATARI games. Best of all, they're all waiting for you at your nearest computer/software or game store — today!

Our games are covered under a 14-day "satisfaction or your money back" guarantee.

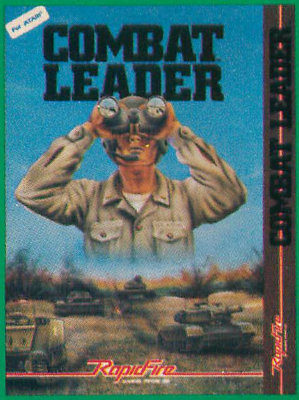
1.

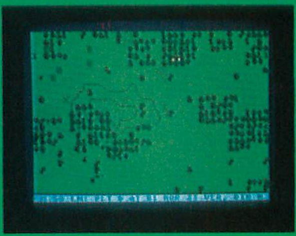




This historical simulation of four great World War II carrier battles is so thorough and realistic, it takes into account every last ship and plane! It is one monster of a game in scale, yet it is so easy to play.
On 40K disk.
$59.95

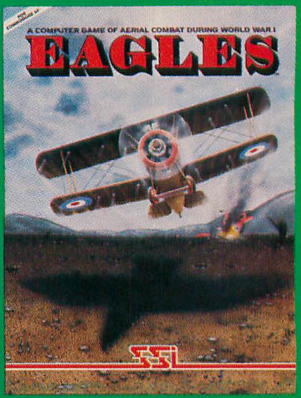
2.

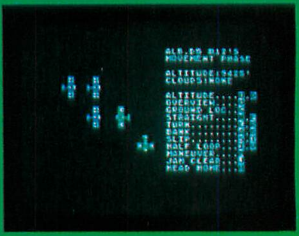




Here's a real-time wargame that gives you the speed of arcade games and sophistication of strategy games. It is one of the finest and fastest tank-battle games ever made. You can choose from over 70 tanks, from the Nazi Tiger to the Abrams M-1.
On 40K disk & 40K cassette.
$39.95

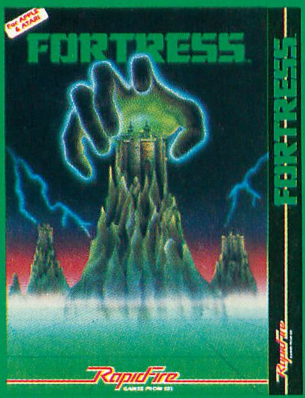
3.

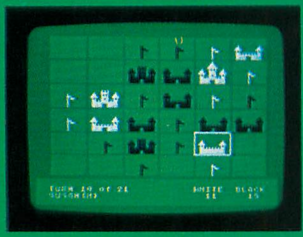




Take to the skies with your trusty Sopwith Camel or Fokker DR-1 or any number of World War I planes. You can set up duels between two planes or stage grand dogfights involving as many as 20 planes.
On 40K disk.
$39.95

4.

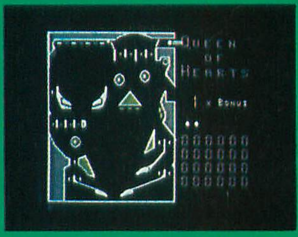




A simple and fast strategy game of power and conquest that's right up there with the classics such as chess. When you play the computer, it actually learns from you and improves with time to become the ultimate opponent.
On 40K disk.
$34.95

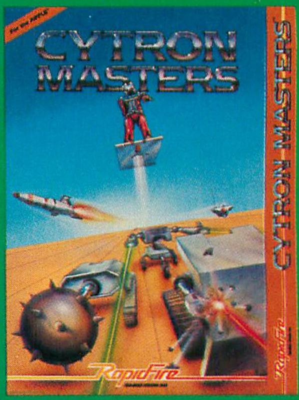
5.





Finally, a true video pinball game for all you pinball fanatics. It boasts all the features of a real pinball game – from sequential play right down to the bells and buzzers. We must forewarn you: it is totally addictive.
On 48K disk.
$34.95

6.





The army of the future is comprised not of men but of machines: the Cybernetic Electronic Devices. They are deadly but mindless. It is up to you as the Cytron Master to lead them to victory in this arcade-style game.
On 48K disk, $39.95; and 32K cassette, $34.95.

Screen displays are not necessarily from the Atari®. Games are for the Atari® 400/800/1200.

SSI

STRATEGIC SIMULATIONS INC

If there are no convenient stores near you, VISA & Mastercard holders can order direct by calling **800-227-1617, ext. 335 (toll free).** In California, call 800-772-3545, ext. 335.

To order by mail, send your check to: STRATEGIC SIMULATIONS INC, 883 Stierlin Road, Bldg. A-200, Mountain View, CA 94043. (California residents, add 6.5% sales tax.)

WRITE FOR A FREE COLOR CATALOG OF ALL OUR GAMES.

SCREEN DUMP

SDUMP your graphics screen to a printer

by JERRY WHITE and FERNANDO HERRERA

REQUIRES 32K RAM AND DISK, PRINTER

Many ANTIC readers have asked for a utility program that would dump a **Micro-Painter** screen to an Epson-Graftrax Plus printer. The following SDUMP program will do just that. All you need is a Micro-Painter-generated disk file and an Epson MX or FX printer equipped with Graftrax Plus.

One problem that generally causes printed pictures to lose graphic appeal is a lack of color. Most Micro-Painter screens contain four colors, and most printers can only put black print on a white background. The SDUMP program provides a compromise by adding the capability to print two different shades of grey.

Look at lines 320 through 390. I have them set up to leave the background in white, print color register 2 in dark grey, color register 1 in a lighter grey, and color register 0 in black. Depending on the requirements of your picture, you may want to change these "color" assignments. Swap line numbers 330 and 390, and the background would print in black and color register 0 would be white. Likewise, swapping line numbers 350 and 370 would reverse the density of the grey shades in color registers 1 and 2.

SDUMP could also be used to dump a Graphics 24 screen with minor adjustments. Line 660 sets up the Antic Mode 14 display list. By removing "ASM = USR(ADR(DL$))" from that line, you could dump a Graphics 24 screen.

The routine at line 670 opens the disk file and reads it into screen memory. The parameters being passed through the USR call are the address of the machine-language subroutine, the IOCB number, the command (7 = read), the address at which to begin storing data, and the number of bytes to move.

If you are going to try to read a picture file that is not generated in Datasoft's format, you may have to change or delete lines 690 through 720. Micro-Painter files contain color bytes at the end of the disk file. Other graphics packages may store these bytes at the beginning of the file, or not at all. To use SDUMP with files other than Micro-Painter files, you'll have to adjust accordingly.

SDUMP contains no error trapping, so you'll have to be careful. Be sure to SAVE the program before using it for the first time, because one incorrect byte of data could cause your computer to lock-up. If this happens, reboot, then reload the program and double-check your DATA statements.

Make sure that your printer is ready, and that you know the filename of your disk file before you begin. When you enter your disk filename, make sure that you include the correct device and filename, as well as the extension, if any.

Any other requests? It's difficult to know what kind of programs you would like to see in ANTIC unless you tell us. If you have any questions on Atari BASIC or suggestions for other utility programs, please send them to Jerry White in care of ANTIC Magazine.

```
94 REM ***SCREEN DUMP****
95 REM *BY JERRY WHITE &*
96 REM *FERNANDO HERREA *
97 REM * ANTIC MAGAZINE *
98 REM * JANUARY, 1984  *
99 REM ******************
```

continued on next page

```
100 REM SDUMP 9/13/83
110 GRAPHICS 0:POKE 82,4:POKE 752,1:POKE 710,160:POKE 712,24:GOSUB 160
115 REM SETUP PRINTER FOR GRAPHICS
120 CLOSE #1:OPEN #1,8,0,"P:":? #1;CHR$(27);CHR$(64);CHR$(27);CHR$(51);CHR$(24)
130 PR1=ADR(PR1$):PR2=ADR(PR2$)
135 REM SCREEN TO PRINTER LOOP
140 FOR X=0 TO 39:PICT=191*40+SCR+X:STR=USR(1536,PICT,PR1,PR2):? #1;PR1$:? #1;PR2$:NEXT X
150 CLOSE #1:GRAPHICS 0:? :? "BASIC":? "IS";:END
155 REM SETUP SUBROUTINES
160 DIM PR1$(964),PR2$(964),HD$(4),FN$(15),CIOV$(42),DL$(33)
170 ? :? :? "SDUMP CREATING:":LIST 175
175 REM PRINT STRINGS
180 FOR ME=1 TO 4:READ IT:HD$(ME,ME)=CHR$(IT):NEXT ME
190 DATA 27,76,192,3
200 PR1$=CHR$(0):PR1$(964)=CHR$(0):PR1$(2)=PR1$:PR2$=PR1$
210 PR1$(1,4)=HD$:PR2$(1,4)=HD$
220 REM DISK TO SCREEN SUBROUTINE
230 LIST 220:FOR ME=1 TO 42:READ IT:CIOV$(ME,ME)=CHR$(IT):NEXT ME
240 DATA 104,104,104,10,10,10,10,170,104,104,157,66,3,104,157,69,3,104,157,68,3
250 DATA 104,157,73,3,104,157,72,3,32,86,228,169,0,133,213,189,67,3,133,212,96
260 REM DISPLAY LIST SUBROUTINE
270 LIST 260:FOR ME=1 TO 33:READ IT:DL$(ME,ME)=CHR$(IT):NEXT ME
280 DATA 104,173,48,2,133,203,173,49,2,133,204,160,1,200,177,203
290 DATA 201,15,240,4,201,79,208,4,233,1,145,203,192,200,208,237,96
300 REM COLOR REGISTER PATTERNS
310 LIST 300:FOR ME=1731 TO 1746:READ PAT:POKE ME,PAT:NEXT ME
320 REM BACKGROUND=WHITE
330 DATA 0,0,0,0
340 REM COLOR REG.2=DARK GREY
350 DATA 10,5,10,5
360 REM COLOR REG.1=LIGHT GREY
370 DATA 10,0,5,0
380 REM COLOR REG.0=BLACK
390 DATA 15,15,15,15
400 REM SCREEN DUMP SUBROUTINE
410 LIST 400:FOR ME=1536 TO 1730:READ ML:POKE ME,ML:NEXT ME
420 DATA 104,104,133,204,104,133,203,104,141,192,6,104
430 DATA 141,191,6,104,141,194,6,104,141,193,6,169
440 DATA 1,133,207,169,191,133,208,160,0,177,203,141
450 DATA 190,6,165,207,240,28,169,0,133,207,173,193
460 DATA 6,24,105,4,141,193,6,133,205,173,194,6
470 DATA 105,0,141,194,6,133,206,76,95,6,169,1
480 DATA 133,207,173,191,6,24,105,4,141,191,6,133
490 DATA 205,173,192,6,105,0,141,192,6,133,206,32
500 DATA 137,6,32,146,6,32,160,6,32,137,6,32
510 DATA 169,6,32,160,6,165,207,240,177,56,165,203
520 DATA 233,40,176,2,198,204,133,203,198,208,165,208
530 DATA 201,255,208,151,96,173,190,6,41,3,10,10
540 DATA 170,96,160,0,189,195,6,145,205,232,200,192
550 DATA 4,208,245,96,173,190,6,74,74,141,190,6
560 DATA 96,160,0,189,195,6,10,10,10,10,24,113
570 DATA 205,145,205,232,200,192,4,208,238,96,0,0,0,0,0
590 GRAPHICS 0:POKE 710,208:POKE 712,24:? :? ,"SCREEN DUMP"
600 ? :? " MICROPAINTER TO EPSON GRAFTRAX+"
610 ? :? "by Jerry White & Fernando Herrera"
620 ? :? "Print intensity patterns may be"
630 ? "altered by swapping data in the"
640 ? "following lines:":LIST 320,400
650 POKE 752,0:POKE 82,2:? :? "ENTER DEV:FILENAME.EXT";:INPUT FN$
655 REM SETUP ANTIC 14 DISPLAY LIST
660 GRAPHICS 24:SCR=PEEK(88)+PEEK(89)*256:ASM=USR(ADR(DL$))
670 CLOSE #1:OPEN #1,4,0,FN$:ASM=USR(ADR(CIOV$),1,7,SCR,7680)
680 REM GET MICROPAINTER COLOR BYTES
690 GET #1,IT:POKE 712,IT
700 GET #1,IT:POKE 708,IT
710 GET #1,IT:POKE 709,IT
720 GET #1,IT:POKE 710,IT
730 CLOSE #1:RETURN
```

TYPO TABLE

Variable checksum = 391188

Line num range	Code	Length
94 - 135	KT	451
140 - 220	PG	503
230 - 320	WO	513
330 - 440	XD	359
450 - 550	JB	502
560 - 660	FC	540
670 - 730	RT	244

TYPER'S TOOLBOX

[RETURN] to paradise.

by STEVEN BOBULSKY

In November of 1981 I brought home my new Atari 800 computer and experienced a major lifestyle change. It was the end of getting to bed early! I was very proud of my new pet that performed tricks and played games. My spare hours went to hunting in computer shops for anything that contained an article or program written for or about the Atari.

Fifteen months later, with 32K of extra RAM, a disk drive, an 850 interface, a printer and a subscription to ANTIC, I'm doing things with my 800 that I really never dreamed of when I bought it: data files, word processing, mailing lists, and, of course, great games. For many of these applications, commercial software is fine, but I have found that typing in public domain programs from magazines is not only rewarding, it's practically free!

Typing, however, is time consuming. And typing BASIC programs into the computer calls for a skill that even many typists are not really good at — typing numbers. Numbers aren't so tough, if they are sprinkled throughout the text of the program, but when you get to number-packed DATA statements, the going gets harder. And the best games (and some other types of programs) contain a good many DATA statements for such things as machine-language subroutines for super-quick graphics and for loading new character sets. It seemed to me that typing DATA statements would be a lot easier if you could just type the numbers and press the [RETURN] key. The **Typer's Toolbox** is designed to do exactly that.

Fr. Steven Bobulsky is an associate Pastor at a Chicago parish. He is busily working on his Atari to "computerize" his church's day-to-day operations.

USING TYPER'S TOOLBOX

When you run the program, the line writer asks for the beginning line of the statements that you want to type. Next, it asks for the increment between line numbers (when you're lucky, the program lines are evenly spaced, but if they aren't, you can use a value of "1" for the increment, and just skip over line numbers that you don't need). Then, it asks you to specify if you are writing DATA statements or another type of statement.

If you are entering "regular" program lines (which for our purposes means anything except a line that starts with the word "DATA"), Toolbox will automatically provide you with a line number and will position the cursor. Type the statement as usual. When you have finished typing the line, press [RETURN]. The program will display the line you have typed and will also display the next line (in case there is an entry on it already). If you press [RETURN] again, the process will be repeated. Any other key will take you to a function menu. Prompts are provided at every point in the program to make your choices clear.

When you are writing DATA lines, the screen will display the line number to which you are writing and request you to input the elements one by one. If you accidentally press the [RETURN] key without making an entry, the program will give you one more chance to make an entry into that position. When you are finished with a line, you must press the [RETURN] key *twice* to process the line. Toolbox will write the line number, the word "DATA," and supply all or any commas that the line might need. All you ever have to do is type the material that is on the DATA line, whether numbers or words, and press the [RETURN] key! When the DATA statements consist exclusively of numbers in a long list, which is frequently the case, this method dramatically speeds up the process of copying.

MENU OPTIONS

The menu presents several options after you've finished typing each line. You may (1) continue, (2) edit the last line that you typed, (3) change the increment between the lines that you are writing, (4) shift modes between "regular lines" or DATA lines, (5) restart without losing any lines, or (6) initiate a line-delete utility. If you are writing DATA lines, the number of elements you have entered on the last line, and the total number of elements you have typed into any or all DATA lines, will be displayed. At the bottom of the screen, the current mode of line

continued on next page

entry and the amount of free memory are displayed. Pressing any of the numbers indicated on the menu automatically executes that function.

You may change the line increment in any way, going either forward or backward to any line number. However, if you step backward to write a line, the computer will write that line and again begin stepping *forward* from that point.

The "line-delete utility" can be used to delete *any* range of lines. This utility writes commands to the screen, then exits the Toolbox program completely to execute those commands in the computer's direct mode. This is accomplished by using the program to write all instructions necessary for the delete process on the screen, and then asking the cursor to pass over those lines as it exits the program.

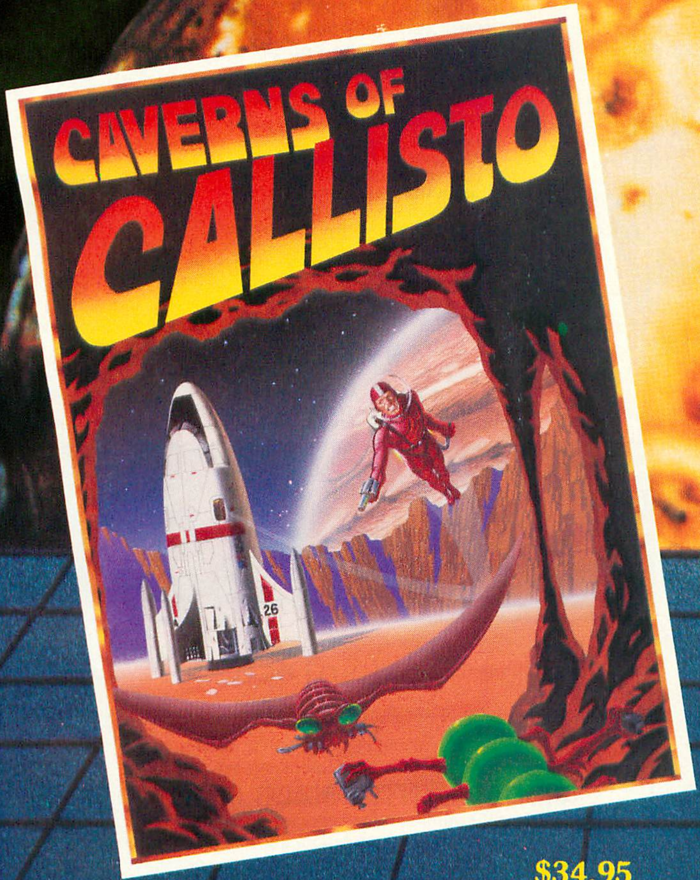
CRUNCHED CODE

To run Toolbox, type it in and LIST the lines to the disk or tape. Whenever you want to use the program, enter it into memory and then give the command "GOTO 30000." Using this program has cut the time it takes me to enter blocks of DATA statements by two-thirds! Toolbox runs exactly the same in both Atari BASIC and BASIC A+.

```
29995 REM ***TYPER'S TOOLBOX**
29996 REM *BY STEVEN BOBULSKY*
29997 REM *  ANTIC MAGAZINE   *
29998 REM *  JANUARY, 1984    *
29999 REM ********************
30000 CLR :TRAP 30000:? "↖Enter the starting line number.":INPUT L:? :? "Increment between lines";:INPUT I
30010 DIM Y$(120),Z$(120):T=0:IF L>32699 THEN 30000
30020 TRAP 30020:? "↖Enter '1' for regular lines":? "OR '-1' for DATA lines.":INPUT A:IF ABS(A)<>1 THEN 30020
30030 E=0:IF A=1 THEN GOTO 30310
30040 X=1:C=0:I=ABS(I):? "↖DATA MODE:":? :? "Enter your items, or none,& hit RETURN twice when finished."
30050 ? :? L;" Item #";X;:INPUT Z$:? :Y$(LEN(Y$)+1)=Z$:Y$(LEN(Y$)+1)=",":X=X+1:C=C+1
30060 IF Z$<>"" THEN E=0:GOTO 30050
30070 IF Z$="" THEN Y$(LEN(Y$))="":X=X-1:C=C-1:E=E+1:IF E=2 THEN 30090
30080 IF E=1 THEN 30050
30090 IF Y$="" THEN 30120
30100 T=T+C:Y$(LEN(Y$))=""
30110 E=0:? "↖":? :? L;"DATA";Y$:? :? :? "CONT":POSITION 0,0:POKE 842,13:STOP
30120 GOSUB 30300:LIST L:? :? "Proposed next line is ";L+I:? :LIST L+I:? :? "RETURN=continue":?
30130 ? "Any other key = MENU":CLOSE #1:OPEN #1,4,0,"K:":GET #1,K:CLOSE #1:IF K=ASC(CHR$(155)) THEN 30290
30140 GOSUB 30300:LIST L:? :IF A=-1 THEN ? L;:? " item count=";C:? "Items in current block=";T:? :?
30150 ? :? "Current line";:? "         =";L:? "Proposed next line =";L+I:LIST L+I:? :? "1=next line":? "2=edit"
30160 ? "3=change increment":? "4=shift modes":? "5=restart (no lines lost)":? "6=Delete a line range."
30170 ? :? "MODE:";:IF A=1 THEN ? "Regular lines.":? :? "Free RAM=";FRE(0)
30180 IF A=-1 THEN ? "DATA lines":? :? "Free RAM=";FRE(0)
30190 IF FRE(0)<400 THEN ? "MEMORY CRITICAL":? "LIST or SAVE your program lines":? "to tape or disk.":END
30200 CLOSE #1:OPEN #1,4,0,"K:":GET #1,K:IF K<ASC("1") OR K>ASC("6") THEN 30200
30210 IF K=ASC("2") THEN ? "↖EDIT:When done, hit RETURN over CONT.":LIST L:?
30220 IF K=ASC("2") THEN ? :? L;" ";:IF A=-1 THEN ? "DATA";" ";
30230 IF K=ASC("2") THEN ? Y$:? :? :? "CONT":POSITION 0,0:STOP
30240 IF K=ASC("2") THEN ? "↖":Y$="":LIST L:? :GOTO 30150
30250 IF K=ASC("3") THEN ? "↖Enter the new line increment.";:INPUT I:? "↖":? :LIST L:Y$="":? :GOTO 30150
30260 IF K=ASC("4") THEN ? "↖":? :? :LIST L:Y$="":X=1:A=-A:GOTO 30150
30270 IF K=ASC("5") THEN GOTO 30000
30280 IF K=ASC("6") THEN GOTO 30390
30290 Y$="":? "↖":X=1:L=L+I:I=ABS(I):GOTO 30030
30300 POKE 842,12:? "↖":RETURN
30310 ? "↖REGULAR MODE":? "Type in the line or just press":? "RETURN to skip to the MENU.":? :? L;" ";
30320 CLOSE #1:OPEN #1,4,0,"K:":GET #1,K:Y$(LEN(Y$)+1)=CHR$(K)
30330 IF K=ASC(CHR$(155)) THEN GOTO 30350
30340 PRINT CHR$(K);:GOTO 30320
30350 IF Y$=CHR$(155) THEN ? "↖No new entry on line ";L:I=ABS(I):GOTO 30150
30360 IF K=ASC(CHR$(155)) THEN GOTO 30370
30370 ? :? :? "CONT":POSITION 0,2:POKE 842,13:STOP
30380 GOSUB 30300:I=ABS(I):LIST L,L+I:? :? :GOTO 30120
30390 ? "↖RANGE DELETE UTILITY:CAUTION":? "ANY line range may be specified, even this program, if you wish."
30400 TRAP 30400:? :? " LOWEST line to delete";:INPUT Q:? "HIGHEST line to delete";:INPUT U:IF Q>U THEN 30400
30410 ? :? "Enter the line increment.":? "(If unsure, use '1' for this value)":INPUT S:IF Q>32700 THEN 30400
30420 ? "↖Lowest line  =";Q:? "Highest line =";U:? "ARE YOU SURE?":? :? "Press 'Y' to continue,":POKE 702,64
```

continued on page 78

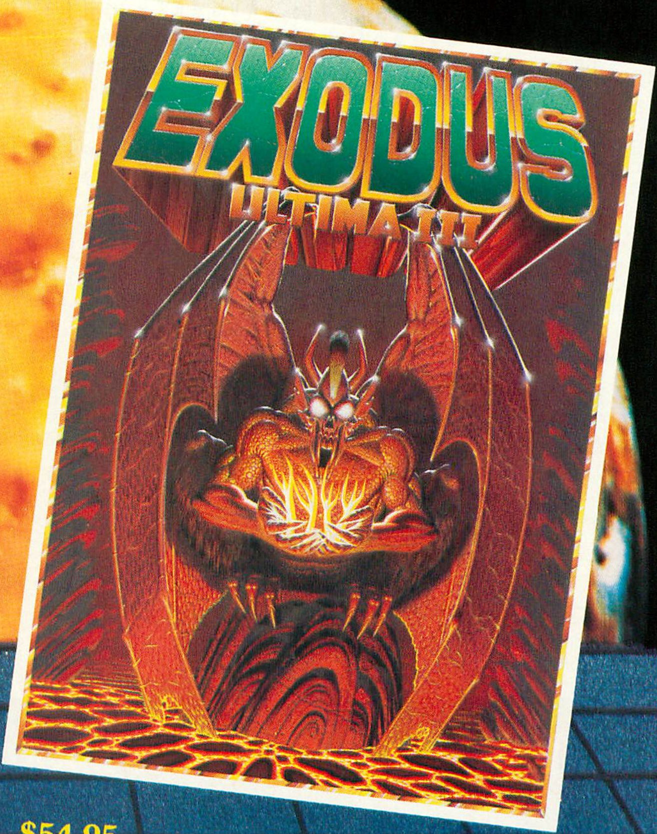
Two Great Games From Out of This World!



$34.95

CREATED BY CHUCKLES

From the depths of Callisto, Origin brings you the excitement of arcade action with the intrigue of adventure gaming. Real time vertical and horizontal scrolling, along with expert color graphics, offer a truly unique entertainment experience. Chuckles has created five levels of caverns filled with treacherous monsters, boiling volcanoes, and secret passages that will challenge even the most experienced player.



$54.95

CREATED BY LORD BRITISH

From the land of Sosaria, EXODUS: ULTIMA III sets a new standard for excellence in fantasy role-playing games. Continuing the tradition of innovation established by Lord British, EXODUS features new animated graphic techniques, full sound effects, and mult-character interplay. Step into our new fantasy world and use your skill and wit to find and destroy the insidious EXODUS!

MOCKINGBOARD ENHANCED

Available for Apple II/II+/IIe and for Atari 400/800

from your local dealer or direct from:

ORIGIN SYSTEMS INC.

18100 Upper Bay Road, Suite 202

P.O. Box 58009
Houston, Texas 77258
(713) 333-2539

ORIGIN SYSTEMS - ORIGINATORS OF EXCELLENCE

TYPER'S TOOLBOX continued from page 76

```
30430 ? "or any other key to abort.":C
LOSE #1:OPEN #1,4,0,"K:":GET #1,K:IF K
<>ASC("Y") THEN ? "↖":GOTO 30150
30440 ? :? "If you want to watch the p
rocess,":? "press '1'.  If you would l
ike speed,"
30450 ? "any other key will blank the
screen":? "during the delete to initia
te":? "'direct memory access.'"
30460 CLOSE #1:OPEN #1,4,0,"K:":GET #1
,K:? "↖":IF K<>ASC("1") THEN POKE 559,
0:CLOSE #1
30470 POSITION 2,11:? "POKE 842,12:POK
E 559,34:CONT":POSITION 2,4:? "POSITIO
N 2,6:? Q:POSITION 2,4"
30480 POSITION 2,7:? "Q=Q+S:IF Q<=U TH
EN POSITION 2,1":POSITION 2,0:POKE 842
,13:STOP
30490 ? "↖":TRAP 30000:POKE 559,34:GOT
O 30150
```

TYPO TABLE

Variable checksum = 123395

Line num range	Code	Length
29995 - 30040	MD	539
30050 - 30130	SW	580
30140 - 30190	MS	537
30200 - 30280	FJ	503
30290 - 30380	WX	519
30390 - 30430	CZ	529
30440 - 30490	HE	544

THE GREATEST ATARI® GAME OF ALL TIME.

Object: Capture more programs than from any other source.
Score: The best prices for programs win.

Actually, every member of CompuClub is a winner, because no one pays lower prices for Atari® programs than our members.

And no other source offers as many programs, with a descriptive catalog covering every piece of software we offer!

CompuClub has hundreds of Atari® programs: games, education and business. It's an astonishing selection, but just as incredible are our prices and our catalog.

Prices always at least 25% below retail

Fact is, usually our prices on selected programs are even lower than that. And we're always running sales with savings of at least 50% from the list price on some of your favorite programs.

And there are several ways to pay: MasterCard, Visa, or the ever popular check or money order.

Annotated catalog, updated every 45 days

We don't like to knock the competition, even by implication, but there are definite advantages to a CompuClub membership. One of the best of them is our catalog, which not only offers an unbelievable number of programs, but includes a description of each and every one of them.

In fact, the catalog is so big and filled with so much information that we print a separate price list. And we keep adding so many programs that we have to update the catalog nine times a year during the course of your membership.

Think of it! No more wondering what's behind the fancy label, no more shooting in the dark or depending on the word of a clerk who may not be familiar with the program you're interested in.

Exclusivity for Atari® owners

We're not trying to be all things to all people. We are definitely the best thing since the floppy disk for Atari® owners, and only Atari® owners. CompuClub is very exclusive.

Our catalog is thick with hundreds of Atari® programs, and only Atari® programs. No more fumbling your way through thickets of strange symbols and codes for the different computers everybody else's catalogs try to cover.

The rules

The rules are simple. To play the CompuClub "game," you've got to be a member. Anyone can join, as long as they fork over a mere five bucks, and agree to buy three programs during the year of membership. Order and buy your programs at any time during the year, but we're sure with our selection and prices that you'll want to get going right away.

Five dollars buys you a one-year membership, exclusive Atari® program offerings, a fully annotated catalog with 9 updates during the year, a current price list (and any necessary updates), a subscription to our newsletter, and discounted prices at all times, including announcements of our periodic sales, which feature savings of 50% off and more.

You can join by filling out the coupon below, or get an instant membership by calling our toll-free number and giving us your MasterCard or Visa identifying number. We'll forward your registration and sales order materials, your catalog and price list, so you can begin ordering right away.

That's all there is to it.

If you think you play a good game, you've got to join CompuClub. It's easy to play, and it saves you money.

We think it's the best Atari® game going, because with us, everybody wins.

CompuClub
Where Atari owners belong.



YES, **CompuClub,** PO Box 652, Natick MA 01760 **ATTN: Dept. AT**
I want to play the greatest Atari® game of all. Please enroll me in CompuClub for one year, and rush my registration material, catalog and price list to me. I agree to buy any three programs during the course of my year of membership. I have enclosed my payment of $5, or authorization to bill my charge card account.

Please make your check payable to CompuClub
Payment enclosed ☐ check ☐ money order.
Bill my ☐ MasterCard ☐ Visa ID #__________

Expires__________



Signature__________

Name__________

Address__________

City______ State______ Zip______

Computer Model__________ Disk ☐ Tape ☐

If you are not satisfied in any way with the conditions of your membership, you may cancel before ordering any programs from CompuClub. Your membership fee of $5 will be cheerfully refunded.



To join by phone, call toll-free.

800-631-3111

Please have your MasterCard or Visa card ready when you call. In Massachusetts, call 1-617-879-5232. We will forward your membership materials after confirming authorization for your charge.

microscreens

This month's microscreens were submitted by Guy Conrad, a very artistic fellow who paints, sculpts and does special-effects makeup for amateur films. The artist lives in Woodside, Calif., and works for Dynaloc Corp. He used his Atari and **Micro-Painter** (DataSoft) to create these microscreens.



Guy's portrait of *Janet*, with its bold outlines, cascading hair, and a very contented-looking cat, is a surprisingly realistic microscreen. Strong outlines and shadows seem to mark Guy's work.



Rose also reveals Guy's knack for vividly recreating natural beauty. Again, bold outlines and a masterful use of shading and shaping are displayed.



Falls shows another side of Guy's art. This scene's lush foliage and rushing water conjure up a sense of being enveloped by nature.

"Microscreens" is a monthly feature of ANTIC that provides a showcase for Atari-generated art created by our readers. Anyone who is interested is encouraged to submit one or more microscreens (on disk) for consideration, along with a short biographic sketch and any necessary instructions on how to load or run the picture. Commercial graphics-utility programs may be used, but need not be. Please identify any utility programs that you use to create your microscreens. And keep those microscreens coming in.
—ANTIC ED

If you are serious about personal finance...

- **Budget Forecast** – 26 expense categories
- **Check Entry** – easy data entry – scan & modify 26 major & 36 sub-categories – information block
- **Check Search** – single or multiple parameters – (up to seven) to search entries
- **Tabulations** – detailed expense vs. budget comparisons by month, year-to-date, category
- **Bar Graphs** – screen displays in graph form expenses vs. budget – by month or category – printing with graphic capable printers
- **Check Reconciliation** – fast clearing of resident checks & deposits, complete summary report
- **Checkwriter** – prints your custom checks
- **Printouts** – most popular printers
- **Multi-Colored Graphics • Audio Enhancements**
- **7 Utility Programs • User-Friendly Operation**
- **Easy To Use Instruction Manual • Audit Report**
- **Handsome Tinted Plastic Storage Case**



COMPUTE!
The Leading Magazine Of Home, Educational, And Recreational Computing

In a feature editorial.

"If you want to use a finance system, but don't want to spend several days trying to learn how to use one, then A Financial Wizard by Computari may be just what you need."

"The illustrated manual that comes with this program is clear, direct, and very thorough."

"It appears that this finance system was designed to achieve the best and most comfortable working relationship between the user and the program."

"The check entry routine is the most attractive feature of this finance system. Data prompts are very clear and the category item names are displayed at all times during data entry for your convenience."

"The file search capabilities of this program are superior. You are offered seven ways to look up the checks."

"The system is disk intensive. All data is saved automatically and immediately following all routines that either enter data or modify it."

"Scanning your entries is made possible by pressing START. You can see records very quickly this way."

"This is an excellent finance system – entertaining, accurate, and fun to use."

ANALOG COMPUTING
THE MAGAZINE FOR ATARI COMPUTER OWNERS

Analog Magazine in a comprehensive study of personal finance systems for Atari* computers.

"A Financial Wizard from Computari is by far the best of these programs and will be the standard of comparison for the others."

"The check entry mode is easy to use..."

"The way a Financial Wizard handles your tabulations is excellent. You can chart your actual expenses vs. your budget by month, by category or year to date."

"...where it really outshines the rest is in the check reconciliation."

"In effect it gives you your bank statement on the screen, a complete list by month of all your checks and deposits."

"A Financial Wizard has one disk that does everything..."

"Graphics, while really not a factor in the quality of programs of this type, do make your budgeting chores a little more pleasant. Again A Financial Wizard comes out on top."

"Everything about this program is excellent..."

Antic
The ATARI Resource

In a Report from Antic.

"Like most Atarians, I am captivated by the graphic, color and sound capabilities of my machine. Nothing quite discourages me more than to boot up an applications program (personal, business, etc.) and to be presented with the standard graphic 'o' white characters on a blue screen.

Of course the usefulness and effectiveness of a program is of primary importance. However, enhancing the dullest of applications programs with some of Atari's charms, is a great asset. A Financial Wizard, a personal finance program by Computari's Bill McLachlan, is an excellent example of an applications program that integrates many of the Atari's features into a well conceived and executed program."

"The use of color and sound in the data input prompts and error checking routines are so well done that it's quite simple to boot up the disk, follow along with the very clear documentation, and be 'up and running' in short order."

"I give A Financial Wizard high marks in ease of use, documentation and performance. If a disk-based home finance package is in your future, The Wizard should get serious consideration."

Computari's
A Financial Wizard 1.5
The logical choice.

The system is designed for Atari computers having a minimum of 32K and operating from a disk drive. The cost is only $59.95 plus $3 for handling/postage.

If your dealer does not have **A Financial Wizard** . . . Telephone orders are accepted on Mastercharge or Visa credit cards. Mail order must be accompanied by check or money-order or credit card #.

Dealer inquiries invited.

*trademark of Atari Inc.

OK residents ad 4% Tax.

Exclusively thru **ON LINE** COMPUTER CENTERS of OKC™ (405)751-7925

10944 North May, Oklahoma City, OK 73120

CHECK ENCLOSED VISA MASTERCHARGE

NAME (Print) ____________________

Address ____________________

City ____________________

State ____________________ Zip ________

Card # ____________________ Exp. ________

Signature ____________________

A CABLE FOR YOUR PRINTER

The Centronics connection

by ROBERT DEWITT
Managing Editor

To connect your Atari computer to a printer, you will need a cable. If the printer is an Atari printer, the cable will be supplied. A few other printers may also provide a ready-made cable, or make one available at extra cost (Alphacom, for example).

An acceptable alternative is to build your own cable. This is not as hard as it sounds, and can save you a little money.

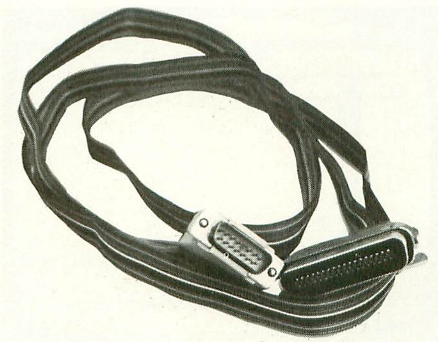
You will need three items and some tools. The items include a 36-pin plug that plugs into the printer, a 15-inch plug that goes into the 850 Interface, and some cable. These can be obtained in any electronics hobby store.

The 36-pin plug you need is standard part number 57-20360. Do not get the similar ICN-365, which, while easier to use, precludes connecting the wires in the desired order.

The 15-pin plug is commonly called a DA-15-P. Do not get the DA-15-S, which is the socket counterpart.

You will also need about five or six feet of multicolored ribbon cable, with a minimum of 12 individually-insulated wires. Extra wires are okay, but these should be clipped so that they do not accidentally short any of the active wires or surfaces. Each wire in the cable has a distinctive color so that you can positively identify the same wire at each end.

Note that the connectors have a trapezoidal shape that prevents incorrect connection. The DA-15-P is arranged this way as you face it:



```
1   2   3   4   5   6   7   8
•   •   •   •   •   •   •   •

  •   •   •   •   •   •   •
  9  10  11  12  13  14  15
```

The Atari uses these pins as follows:

Pin 1 is the Data Strobe
Pin 2 is data bit 0
Pin 3 is data bit 1
Pin 4 is data bit 2
Pin 5 is data bit 3
Pin 6 is data bit 4
Pin 7 is data bit 5
Pin 8 is data bit 6
Pin 9 is not used
Pin 10 is not used
Pin 11 is Signal Ground
Pin 12 is Fault
Pin 13 is Busy
Pin 14 is not used
Pin 15 is data bit 7

Note: data bits are sometimes numbered 1 through 8 in printer manuals.

As you can see, only 12 of the 15 pins are used. At the other end of the cable, only 12 of the 36 pins of the 57-20360 plug are used. These pins are arranged this way as you face the connectors:

```
1                                 18
• • • • • • • • • • • • • • • • • •
• • • • • • • • • • • • • • • • • •
19                                36
```

Connect these pins to the corresponding pins on the other end of the cable:

Pin 1 to the Data Strobe
Pin 2 to data bit 0
Pin 3 to data bit 1
Pin 4 to data bit 2
Pin 5 to data bit 3
Pin 6 to data bit 4
Pin 7 to data bit 5
Pin 8 to data bit 6
Pin 9 to data bit 7
Pin 10 is not used
Pin 11 to Busy
Pin 12–15 not used
Pin 16 to Signal Ground
Pin 17–31 not used
Pin 32 to Fault
Pin 33–36 not used

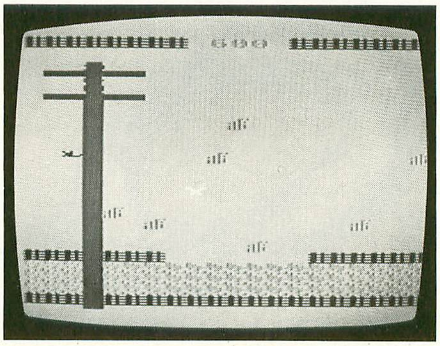
We have successfully used a cable like this to connect the Atari 850 Interface with the following printers: Epson FX100; Gemini 10X; Mannesmann Tally Spirit; NEC PC-8023A; Smith Corona TP1; and Transtar 120. If you have a printer other than these that accepts a 57-20360 plug, verify these connections in the "pin assignment" section of your owner's manual. Chances are that they will be the same, or can be easily modified.

Snowbird

by JOHN WEBER

Requires 16K RAM

"Breaking the ice" is not only a social convention performed by humans — in **Snowbird** it's a physical nightmare, executed by a pack of water rats! The scene is a frozen lake through which the ice-hungry rodentia pick their way. Each time a rat arrives at the surface, it scurries to the nearest water hole, submerges, then gnaws up to the top again. If they aren't stopped soon, the lake will resemble swiss cheese and become a bunch of ice floes — no fun for skaters!



As the Snowbird, you must patrol the lake from atop your telephone-pole perch, and eliminate the pesky vermin. By swooping down and lighting atop a rat, you chalk up 100 points (unless it's a white rat, which is good for 1000). The rats that escape your clutches eat more holes in the ice, and this makes your task more difficult than ever.

Not only do the holes provide a speedy getaway for the rats, but they also become a barrier between you and your prey. If there is a hole between you and a rat, you will find yourself merely flapping your wings until you go around it.

The fire button sends your Snowbird soaring into the sky — and when the button is released, you elegantly dive. Be careful when diving though, because if you touch the water the game ends.

To complete each of the first four levels requires a catch of 20 rats. Higher, harder levels demand only 15. When you've amassed your quota, an ice sweeper erases all the holes, leaving you free to resume your tour of duty as Snowbird, the quintessential pest controller!

List of Variables

HIT — counter for number of rats caught
SCORE — score of game
LOOK — value of new position above or below bird
SEE — value of new position of rat when being moved
HOLE — counter of number of water holes on the screen
NBIRD — location value of bird on screen
HARD — decimal value of level of game
LEVEL — input number by player for level of play

```
10 REM ***** SNOWBIRD *****
20 REM *  BY JOHN WEBER    *
30 REM *  JANUARY, 1983    *
40 REM *  ANTIC MAGAZINE   *
50 REM ********************
100 GOSUB 1000:GOTO 500
400 REM FLY BIRD
410 POKE 77,0:LOOK=PEEK(BIRD-20)
415 IF LOOK=63 THEN 450
420 IF LOOK<>0 OR BIRD-20<SC THEN RETU
RN
430 POKE BIRD,0:POKE BIRD-20,58+64:BIR
D=BIRD-20
440 IF STRIG(0)=0 THEN 400
449 RETURN
450 POKE BIRD,0:POKE BIRD-20,58+64:POK
E BIRD-40,58+64:POKE BIRD-20,63:BIRD=B
IRD-40
459 RETURN
500 ST=STICK(0):POKE BIRD,58+64
510 IF PEEK(BIRD+20)=0 THEN POKE BIRD,
0:BIRD=BIRD+20:POKE BIRD,58+64:POKE 53
760,50:POKE 53761,8
515 POKE 53760,0:POKE 53761,0
520 IF STRIG(0)=0 THEN GOSUB 400
535 NBIRD=BIRD+(ST=7)-(ST=11)
540 IF NBIRD<SC OR NBIRD>SC+479 THEN 5
00
550 IF PEEK(NBIRD)=0 THEN POKE BIRD,0:
```

continued on page 86

If you could live on half your income, just think what you could do with the other half!

Lifestyle Budgeting™...

the first budget planning system that considers **all** your needs.

Others have done it ... so can you.

Dr. Harper Roehm, the author of ***Spending Less and Enjoying It More*** (the McGraw-Hill book that provides the basis for this system), designed **Lifestyle Budgeting** when his income was cut in half due to a career change from corporate auditor to university professor. As a result, his family has lived comfortably, fulfilling their "wants" as well as their "needs" for over 20 years.

We show you "why" as well as "how."

Lifestyle Budgeting is the **first complete budgeting package** to include an easy-to-understand book explaining the behavioral aspects of successful budgeting as well as software to handle the mechanics. It takes a practical approach, showing you not only how you're spending your money, but why ... so you can identify your true priorities and plan for them.

Planning is the key.

Lifestyle Budgeting's software is a forecasting and modeling tool, not a checkbook balancer. It will track your spending patterns, identifying where you're wasting valuable funds. Using that information, and your family's priorities, it will show you specifically how to project and plan future costs.

Little time, and no accounting experience required.

Unlike most budgeting systems, **Lifestyle Budgeting** does not require detailed expense records. It will only take a couple of nights to set up and then only one or two hours a month to monitor. And ... a big plus ... **Lifestyle Budgeting** is written for you, not your accountant.



Use Lifestyle Budgeting on your personal computer.

Since **Lifestyle Budgeting** is something everyone needs, versions are available for IBM, Apple, Commodore, Atari and Coleco personal computers.

Get control of your money today!

Lifestyle Budgeting will be available through your local computer retailer starting in February ... but **why wait?** You can order the complete package today.

Order now!
Call 1-800-547-1565.
In Ohio Call 1-513-435-2335.

Dealer inquiries welcomed.

Yes! I'd like to know how to set up a liveable budget.



Please send me ______ copies of **Lifestyle Budgeting** (including the McGraw-Hill book ***Spending Less and Enjoying It More***, and the accompanying software and manual) at $49.95 each plus $3 for shipping and handling. (In Ohio add 6% sales tax.)

Name ____________________

Address ____________________

City ________ State ________ Zip ________

Type computer ________ Model ________

Check or money order enclosed ☐

VISA or MasterCard No. ____________________

Expiration Date ________ Bank No. (MasterCard) ________

Signature ____________________

Available on diskette only. Allow 4-6 weeks for delivery.

Return coupon and payment to:

culverin Corporation

Culverin Corporation, Lifestyle Budgeting
P.O. Box 503, Centerville, OH 45459.

SNOWBIRD continued from page 84

```
BIRD=NBIRD
551 LOOK=PEEK(BIRD+20)
552 IF LOOK=62 THEN GOTO 850
555 IF LOOK=63 THEN POKE BIRD,0
560 IF LOOK<>187 THEN 580
565 POKE 53761,168:FOR A=250 TO 202 ST
EP -4:POKE 53760,A:NEXT A
570 POKE BIRD,0:BIRD=BIRD+20:RA=0:SCOR
E=SCORE+100:HIT=HIT+1:GOTO 590
580 IF LOOK<>123 THEN 590
582 POKE 53761,170:FOR A=150 TO 250 ST
EP 4:POKE 53760,A:NEXT A
585 POKE BIRD,0:BIRD=BIRD+20:RA=0:SCOR
E=SCORE+1100:HIT=HIT+1
590 IF RND(0)*10>HARD THEN GOSUB 600
595 IF HARD<6 THEN IF HIT=15 THEN POKE
 MOUSE,0:GOSUB 800
597 IF HIT=20 THEN POKE MOUSE,0:GOSUB
800
598 POSITION 9,0:? #6;SCORE
599 POKE 53760,0:POKE 53761,0:GOTO 500
600 REM MOVE RATS
605 IF RA=1 THEN 620
610 RAT=SC+INT(RND(0)*439)+20:IF PEEK(
RAT)<>0 THEN 610
612 IF RAT<SC+141 OR RAT>SC+378 THEN 6
10
615 RA=1:POKE RAT,59+128:POKE RAT+1,63
:HOLE=HOLE+1
618 IF HOLE>50 THEN POP :GOTO 4000
620 IF RAT-1<SC+121 THEN POKE RAT,0:RA
=0:GOTO 610
625 SEE=PEEK(RAT-1):IF SEE=0 THEN POKE
 RAT,0:RAT=RAT-1:POKE RAT,59+128
630 IF SEE=63 THEN POKE RAT,0:RA=0:GOS
UB 650:RETURN
640 IF RND(0)*10>9.5 THEN POKE RAT,59+
64
649 RETURN
650 POKE 53761,136:FOR A=26 TO 116 STE
P 10:POKE 53760,A:NEXT A:POKE 53760,0:
POKE 53761,0:RETURN
800 REM MOVE TRACTOR AND FILL HOLES
810 FOR I=SC+120 TO SC+379
815 SOUND 0,100,18,8:SOUND 1,150,8,8
820 POKE I,57+128:FOR A=1 TO 25:NEXT A
:POKE I,0:? #6;"      ":REM 6 SPACES
825 POSITION 9,0:? #6;SCORE:POSITION 9
,0:NEXT I:HOLE=0
830 HARD=HARD-1:IF HARD<4 THEN HARD=4
849 SOUND 0,0,0,0:SOUND 1,0,0,0:HIT=0:
RETURN
850 REM LOSE BIRD AND START OVER
855 POKE BIRD,0:FOR A=75 TO 180 STEP 5
:SOUND 0,A,10,8:FOR W=1 TO 20:NEXT W:N
EXT A:SOUND 0,0,0,0
860 POKE 53248,0:GOSUB 2050:GOTO 500
1000 REM ENTER LEVEL
1005 GRAPHICS 0:POKE 710,68:POKE 712,6
8:POKE 752,1:POSITION 15,5:? "SNOW BIR
D"
1010 POSITION 3,10:? "WHICH LEVEL DO Y
OU WANT TO START?"
1015 TRAP 1015:POKE 752,0:POSITION 13,
15:? "ENTER 1 - 6 ";:INPUT LEVEL
1025 IF LEVEL<1 OR LEVEL>6 THEN 1010
1030 HARD=10-LEVEL
1050 GRAPHICS 0:POKE 752,1:POSITION 13
,5:? "PLEASE WAIT..."
2000 RAMTOP=PEEK(106)-8:POKE 106,RAMTO
P:CHBAS=RAMTOP:START=CHBAS*256
2005 FOR I=0 TO 1023:POKE START+I,PEEK
(57344+I):NEXT I
2010 FOR I=57 TO 63:MOVE=START+(I*8)
2015 FOR A=0 TO 7:READ SHAPE:POKE (MOV
E+A),SHAPE:NEXT A:NEXT I
2050 GRAPHICS 1+16:POKE 756,CHBAS:POKE
 712,8:POKE 710,0:POKE 709,255:POKE 71
1,34:POKE 708,132
2055 SC=PEEK(88)+256*PEEK(89):BIRD=SC+
45
2060 SCORE=0:HOLE=0:HIT=0
2085 GOTO 3000
2089 DATA 0,0,24,24,24,126,36,126
2090 DATA 0,195,36,24,24,36,0,0
2091 DATA 144,112,113,190,0,0,0,0
2092 DATA 100,255,100,255,100,255,100,
255
2093 DATA 255,255,255,255,0,0,0,0
2094 DATA 192,240,252,255,76,24,1,75
2095 DATA 13,8,42,168,42,170,170,168,4
0
2099 DATA 126,126,255,127,127,254,254,
255,255,126,126,254,255,254,126,126,25
5,255,255,255,126,127,127,126
3000 FOR I=SC+460 TO SC+479:POKE I,60+
192:NEXT I
3005 FOR I=SC+61 TO SC+65:POKE I,61+19
2:NEXT I:FOR I=SC+101 TO SC+105:POKE I
,61+192:NEXT I
3015 FOR I=SC+400 TO SC+459:POKE I,62:
NEXT I
3020 FOR I=SC TO SC+7:POKE I,60+192:NE
XT I:FOR I=SC+13 TO SC+19:POKE I,60+19
2:NEXT I
3022 FOR I=SC+380 TO SC+386:POKE I,60+
192:NEXT I:FOR I=SC+394 TO SC+399:POKE
 I,60+192:NEXT I
3050 REM P/M GRAPHICS
3055 PM=PEEK(106)-16:POKE 54279,PM:PMB
ASE=PM*256:POKE 53277,3:POKE 704,244
3060 FOR I=PMBASE+1024 TO PMBASE+1279:
POKE I,0:NEXT I:RESTORE 2099
3065 FOR I=PMBASE+1024+50 TO PMBASE+10
47+50:READ PMG:POKE I,PMG:NEXT I:FOR I
=PMBASE+1048+50 TO PMBASE+1250
3070 POKE I,255:NEXT I
3095 POKE 53248,72:POKE 623,1:POKE 559
,62
3099 RETURN
```

```
4000 REM DRAW WATER
4007 FOR I=SC+385 TO SC+25 STEP -20
4008 SOUND 0,13,8,15
4010 FOR A=-INT(RND(1)*4) TO INT(RND(1
)*18):POKE I+A,62:NEXT A
4015 FOR A=1 TO 20:NEXT A:SOUND 0,0,0,
0:NEXT I:POKE 712,134:FOR A=1 TO 500:N
EXT A
4025 POKE 712,148:FOR A=1 TO 500:NEXT
A:POKE 53248,0:POKE 710,134:POKE 711,1
34:POKE 709,134:POKE 708,134
4030 POKE 712,134:FOR A=1 TO 500:NEXT
A
4060 GOSUB 2050:GOTO 500
```

TYPO TABLE

Variable checksum = 796062

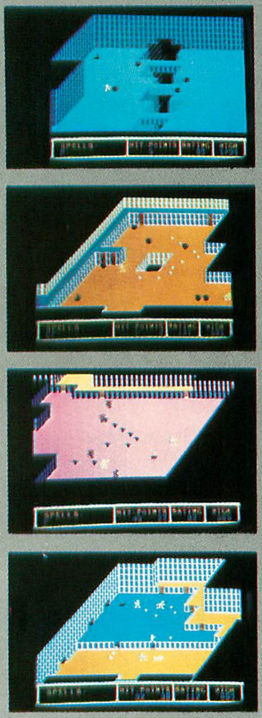
Line num range		Code	Length
10	- 440	PZ	347
449	- 552	CL	479
555	- 598	BK	508
599	- 640	MP	524
649	- 849	KZ	526
850	- 1030	KJ	500
1050	- 2089	JN	513
2090	- 3020	GD	574
3022	- 4008	XB	518
4010	- 4060	GC	380

WHAT IS A TYPO TABLE?

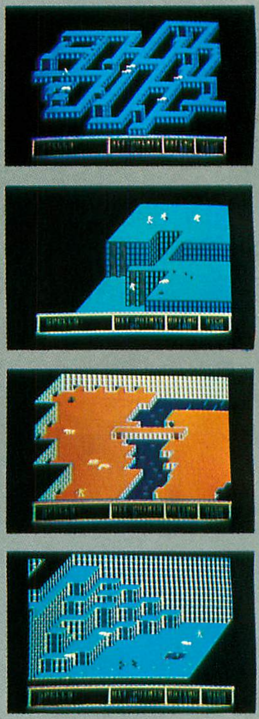
Newcomers to ANTIC may wonder about the "Typo Table" that appears at the end of most of our BASIC listings. TYPO is a program that helps you find typing errors made when entering programs that appear in ANTIC. TYPO will produce a table of values which can be used to pinpoint where an error was made. The TYPO program and instructions originally appeared in Volume 1, Number 3 of ANTIC, and were reprinted in Volume 2, Number 1. The latter issue is still available as a back issue, and the TYPO program itself is included in ANTIC UTILITIES DISK #2.

NOTE: When comparing your TYPO TABLE with the one we publish, first look at the length column. For a given line number range, if your length is only off by one or two, it may be due to spacing. Missing or extra spaces generally occur between quotes or in a REM statement. Spaces must be accurately placed for TYPO to work, so first experiment with the spacing.

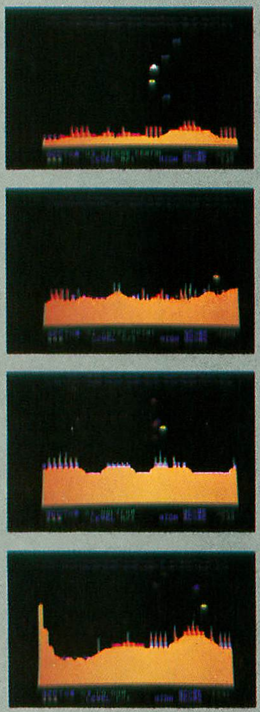
Two for the ATARI* from BRAM Inc. by Mike Edwards

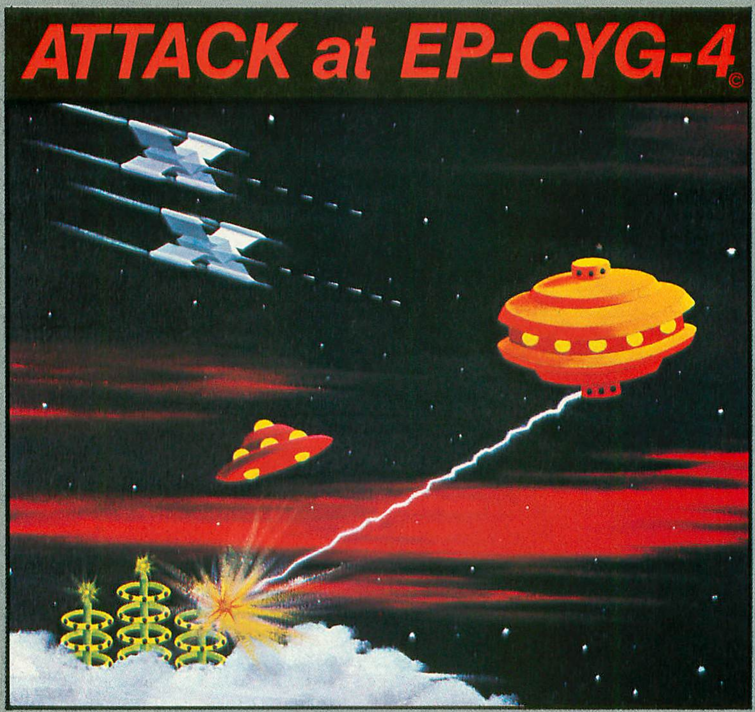


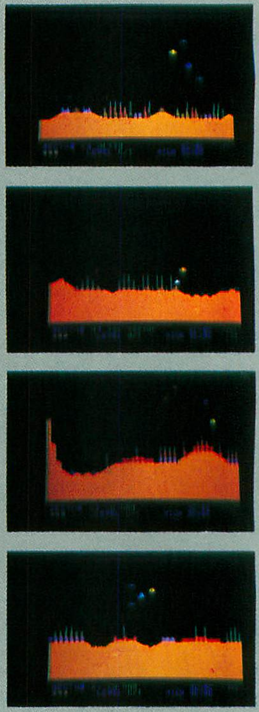




The exciting new arcade hit that features scrolling 3D graphics, on-line instructions, one or two player cooperative, seven different dungeons, 74 different screens, high score save to disk, full sound and color, zombies, poisonous snakes, giant spiders, evil orbs, scrolls, talismans, magic spells, lost crowns and spectacular underground scenery. 48K tape and disk. $34.95







If you're ready for the most challenging arcade space-war game, you're ready for ATTACK at EP-CYG-4. This is an adventure for experienced arcaders. EP-CYG-4 features 3 missions, 130 screens*, Tartillian attack fighters, Cruisers, ground batteries, landscapes, seascapes, Gravitron drive ships, Lyso-blast weapons, advanced joy stick controls, one or two player cooperative.

"I'd recommend EP-CYG-4 to any gamer who is looking for new concepts and challenges in his or her gaming life." D. Duberman, December, 1982 ANTIC

*Disk 32K $32.95; tape 16K $29.95; 2 missions, 70 screens

BRAM Inc.

18779 Kenlake Place N.E. Seattle, Washington 98155 (206) 486-8428

CALLING ALL TAPES

A self-modifying program index for cassettes

by JAMES LUCZAK

The main problem with storing programs on tape is that you must keep a written record of what programs you have and where they are stored. The real problem, though, is where to store the written record so it won't get lost or be destroyed by the kids or the cat.

If you use C-60 or C-90 tape, you'll soon find out that the labels on the tape boxes aren't big enough to hold the names of all the programs that can be put on a long tape, especially if you store a number of short programs on the tape. If you use C-10 or C-15 tape, however, you'll be able to write the program titles you need on the labels that come with the tape box (if a tape box came with the tape!).

Still, you must shuffle through all of your tapes to find the specific program you are looking for at any given time. As a result, tapes are scattered all over the house by the time you find the program you want — and that's assuming the tape you're after hasn't mysteriously disappeared.

GETTING ORGANIZED

The only solution is to get organized. What's needed is an index program that lists all of your programs and provides you with their exact locations. Such a program, it seems to me, must meet the following criteria:

A) It must be easy to use.
B) It must be easy to update with new information.
C) It must have an edit feature.
D) It must check for input errors and explain how to correct them.
E) It must be self-explanatory.
F) It must be unique, so there is no doubt as to what program is running.

James Luczak is a resident of Three Rivers, Michigan, and an avid Forth programmer, as well as a fan of Tape Topics. A communications technician for the phone company, he has written for several computer publications.

A FRAMEWORK FOR INFORMATION

I originally wrote this program to be a header on a C-90 tape, but it can also be stored on a C-10 or C-15 tape. This can then be used as a master-index tape, which means that the names and locations of all of your programs can be stored on a single tape as an index. With a few modifications, this program can also be adapted for a number of other applications, such as an address book, a telephone directory, or a collection of recipes. It's basically a framework for the information you want to store.

ENTERING THE PROGRAM

Because this program modifies itself as it is RUN, it may be a little difficult to get it to RUN correctly the first time you try. A simple typographical error, for example, can cause no end of problems. Here are a few suggestions that may help, though:

1) After you have input the program, SAVE it to a tape. Do not RUN it until you have done this! Then RUN the program and take note of any errors to be corrected. LOAD the original copy. Make any necessary changes, and then SAVE it back to tape as the original. Continue this routine until all errors have been eliminated.

2) When you key in the listing, copy it *exactly* as it is printed in the magazine. Do not change any line numbers or try to compress the program at this time.

The reason that I have asked you to take these precautions is that the program removes specific lines after it is RUN the first time.

HOW TO USE THE PROGRAM

First of all, you *must* make a copy of this program, which should then be stored separately from the version you actually use. This copy should be used as a master copy *only*! The

continued on next page

master will then be used to create index programs for your tapes.

This program must be SAVEd to tape before it is RUN, because the program modifies itself to specifications that you apply. Once you have a copy of the original, RUN the program. Answer any questions that it asks, and then add, delete or edit data as required. Next, CSAVE this version as a header on a C-60 or longer tape, or as a master index on a shorter tape. It will serve as the index to your other programs.

When you want to add new programs, delete or change information, or view the index, CLOAD and RUN the program. The title screen will appear with your tape number and name. Then you will be prompted to view or update. (By the way, be sure to leave extra room on the tape you SAVE this program onto. Since data that is added or changed becomes part of the program, its size will vary.)

ENTERING OR VIEWING DATA

Once you have got the program to work, you will be prompted to enter a tape number. This can be any alphanumeric input of up to three characters. Next, you will be asked if you want to enter a name for the tape. This alphanumeric input can consist of up to twenty characters.

There will be a short pause at this point (of no more than ten seconds or so). Then the title screen will appear with your tape number and name on it, and you will be prompted to either view or update the program.

If you choose to view it, the View/Update/End Menu will appear, because there isn't any data to view yet. If you choose to update, the Edit Menu will appear. This part of the program is very straightforward. It allows you to add, change or delete data.

Meanwhile, the error-protection portion of the program will check to make sure that your entries have been entered using the correct format. It will tell you if you've made a mistake, and, if so, how to correct it.

After you have finished entering data, type DONE. This will be followed by a short pause, as the program adds this data to itself. Furthermore, if you make more than eight entries in a row, a brief pause will follow the eighth entry. This allows the program to automatically add these entries. It then returns you to the data-entry portion of the program. A maximum of thirty entries are allowed.

EDITING YOUR ENTRIES

Once you've finished entering data, the View/Update/End Menu will appear. If you choose view at this stage, your entries will appear in nicely-formatted form. To edit these entries, enter the Edit Mode (Number Two) and use the cursor controls (control arrows) to position the cursor over the entry you want to change. Then type in your change. To delete, follow the same routine and simply type over the text you wish to eliminate.

When you're satisfied with your changes, press [RETURN] and the revised data will be entered into the program. If you LIST the program after you've made some entries, you'll notice that lines 521–529, line 945, line 2005, and lines 2400–2524 are no longer in the program. In addition, line 545 will contain a different number than it did in the original listing. This is why it's so important not to change line numbers in the original program, and to keep a copy of the original listing on a separate tape.

You can, however, always change data or add data to the program once you have entered some data into it. If you want to create a new header tape (for a different tape), you must load the original copy, add your data, and then SAVE it to the tape you want to store it on. You can condense the program, after it has been RUN the first time, but you must be careful not to disturb the line numbers that are to be removed from the program, or lines that employ the RETURN key mode.

TAKE-APART OF PROGRAM

490–520	Dimension strings
521–529	Get tape number and name
540	Clears screen
545	Keeps track of where to add new data lines
560–670	Modifies display list
680–790	Title screen
800–850	Choose update or view
870–910	Reads and displays entries
920–940	Displays view/update/end menu
950–990	Choose from view/update/end menu
1000–1010	Restores to page one of entries
1020–1320	Data lines
1335–1360	Data-edit menu
1370–1400	Choose from data-edit menu
1410–1442	Data-edit instruction screen
1444–1520	Displays and enables data lines to be edited
1590–1680	Enter-program-name screen
1690–1900	Enter-start/end-count screen
1910–2050	Adds data lines to program
2005	Removes lines and itself from program
2015	Updates line 545
2060	Blanks screen while data is being entered into the program
2070–2100	Formats data to be entered into the program
2200–2250	Error-sound routine
2260–2280	Error message
2282–2284	Error message
2300–2310	Loading-data message
2400–2430	Error message
2500–2524	Removes lines from program; adds tape number and name to program.

LINES REMOVE FROM THE PROGRAM

521–529	2400–2430
945	2500–2524
2005	

LINES ADDED TO THE PROGRAM

530
535
1020–?

LINES UPDATED IN THE PROGRAM

545

```
485 REM ***TAPE INDEX****
486 REM *BY JAMES LUCZAK*
487 REM *ANTIC MAGAZINE *
488 REM * JANUARY, 1984 *
489 REM *****************
49Ø ? "↖"
5ØØ DIM A$(84Ø),TP$(38),BL$(38),TL$(3)
51Ø DIM D$(4):D$="DATA":DIM X$(3),Y$(3Ø)
512 DIM B$(4Ø),C$(1),CURSOR$(1)
52Ø BL$=" ":BL$(38)=BL$:BL$(2)=BL$
521 ? CHR$(125):POKE 84,8:? :POKE 85,4:? "ENTER TAPE NUMBER ";:INPUT X$
522 ? CHR$(125):POKE 84,8:? "DO YOU WANT TO NAME TAPE":POKE 84,1Ø:? "EX.(UTILITIES, BACK-UP) ECT."
523 POKE 84,14:? :POKE 85,12:? "(Y OR N) ";:OPEN #1,4,Ø,"K:":GET #1,X:CLOSE #1
524 IF X=89 THEN 527
525 IF X=78 THEN 529
526 GOTO 523
527 ? CHR$(125):POKE 84,8:? :? "ENTER TAPE NAME ":? :? :INPUT Y$
528 IF LEN(Y$)>2Ø THEN 24ØØ
529 POKE 7Ø9,4:GOTO 25ØØ
54Ø ? "↖":POKE 7Ø9,1Ø
545 NULINE=1Ø2Ø
56Ø GRAPHICS 1+16
57Ø POKE 7Ø9,48+8
6ØØ BEGIN=PEEK(56Ø)+PEEK(561)*256+4
61Ø POKE BEGIN-1,71
62Ø POKE BEGIN+2,7
63Ø POKE BEGIN+3,7
64Ø POKE BEGIN+4,48
65Ø POKE BEGIN+5,7
66Ø POKE BEGIN+6,112
67Ø POKE BEGIN+24,65:POKE BEGIN+25,PEEK(56Ø):POKE BEGIN+26,PEEK(561)
68Ø POKE 87,2:POSITION Ø,Ø:POKE 7Ø8,25Ø
69Ø FOR X=Ø TO 19:? #6;"@";:NEXT X
7ØØ POSITION 6,1:? #6;"TAPE   ";X$
71Ø FOR X=Ø TO 19:? #6;"@";:NEXT X
72Ø POSITION Ø,3:POKE 711,152
73Ø CENT=INT((2Ø-LEN(Y$))/2)
732 IF CENT=Ø THEN ? #6;Y$:GOTO 74Ø
734 FOR X=1 TO CENT:? #6;" ";:NEXT X:? #6;Y$
74Ø POKE 87,1:? #6;"do you want to view"
75Ø POSITION Ø,6:? #6;"input  V  for view"
76Ø POSITION Ø,1Ø:? #6;"or do you want to "
77Ø POSITION Ø,12:? #6;"update the tape "
78Ø POSITION Ø,14:? #6;"input  U  for update"
79Ø POSITION Ø,18:? #6;"view or update ?"
8ØØ OPEN #1,4,Ø,"K:":GET #1,X:CLOSE #1
83Ø IF X=85 THEN 1335
84Ø IF X=86 THEN 87Ø
85Ø GOTO 56Ø
87Ø ? "↖":POKE 7Ø9,Ø:POKE 71Ø,246:POKE 712,246:POKE 752,1
88Ø FOR X=Ø TO 14:READ B$
89Ø IF B$="DONE" THEN K=1:GOTO 92Ø
9ØØ ? B$
91Ø NEXT X
92Ø ? :POKE 85,8:? "    V = VIEW NEXT PAGE"
93Ø ? :POKE 85,8:? "    U = UPDATE PROGRAM"
94Ø ? :POKE 85,8:? "    E = END PROGRAM"
942 IF NULINE=132Ø THEN ? :? "DATA BUFFER FULL.CAN'T ADD NEW DATA"
945 IF COUNT=4 THEN COUNT=Ø:? :? "NO DATA TO EDIT YET"
95Ø OPEN #1,4,Ø,"K:":GET #1,X:CLOSE #1
96Ø IF X=86 THEN 1ØØØ
97Ø IF X=69 THEN ? CHR$(125):GRAPHICS Ø:END
98Ø IF X=85 THEN 1335
99Ø ? CHR$(125):GOTO 92Ø
1ØØØ ? "↖":IF K=1 THEN K=Ø:RESTORE 1Ø2Ø:GOTO 87Ø
1Ø1Ø IF K=Ø THEN 87Ø
132Ø DATA DONE
1335 ? CHR$(125):POSITION 12,Ø:? "DATA EDIT MODE"
134Ø POSITION 3,5:? "(1) = ADD TITLES TO TAPE"
135Ø POSITION 3,7:? "(2) = CHANGE OR DELETE TITLES ON TAPE"
136Ø POSITION 3,9:? "(3) = EXIT EDIT MODE"
1365 IF NULINE=132Ø THEN ? :? :POKE 85,1Ø:? "DATA BUFFER IS FULL"
1367 IF NULINE=132Ø THEN ? :? :POKE 85,6:? "NO NEW DATA ENTRIES CAN BE MADE"
137Ø OPEN #1,4,Ø,"K:":GET #1,X:CLOSE #1
138Ø IF X=49 THEN 14Ø2
139Ø IF X=5Ø THEN 141Ø
1395 IF X=51 THEN ? CHR$(125):GOTO 92Ø
14ØØ GOTO 137Ø
14Ø2 IF NULINE=132Ø THEN 1335
14Ø4 GOTO 159Ø
```

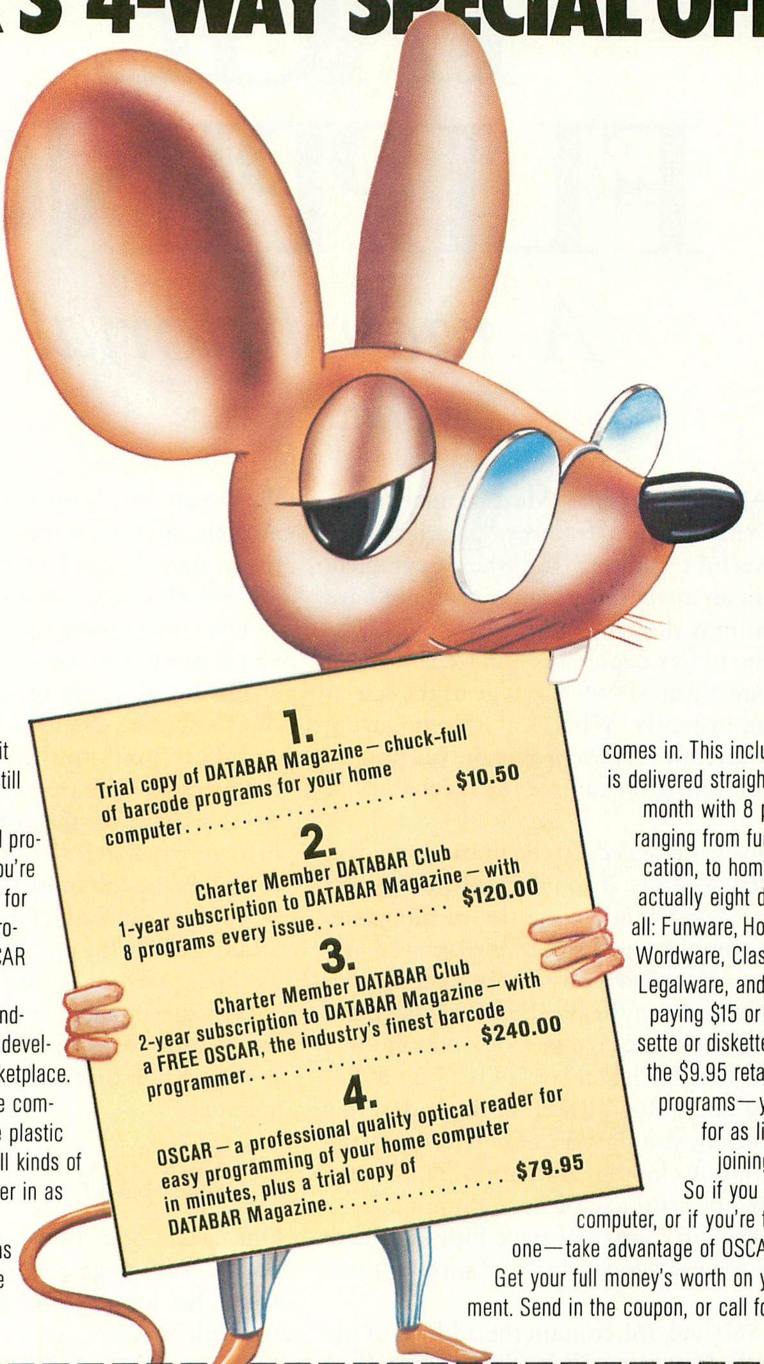
continued on next page

```
1410 ? CHR$(125):POSITION 2,4:? "TO DE
LETE AN ENTIRE LINE,TYPE SPACES  OVER
EVERYTHING EXCEPT THE LINE NUMBER"
1411 ? :POKE 85,13:? "** CHANGES **":?

1412 ? "USE CURSOR CONTROL ARROWS TO P
OSITION CURSOR.THEN TYPE OVER THE ENTR
Y TO BE CHANGED"
1420 ? :? "IF NO CHANGES ARE TO BE MAD
E JUST HIT RETURN AND THE REST OF THE
PAGE YOU   ARE WORKING ON WILL APPEAR"
1430 ? :? "DO NOT HIT RETURN UNTIL ALL
 CHANGES   ARE MADE..REGARDLESS OF LIN
E NUMBERS.."
1442 ? :? "HIT ANY KEY TO CONTINUE ":O
PEN #1,4,0,"K:":GET #1,X:CLOSE #1
1444 POKE 752,0:? CHR$(125):COUNT=0:GO
SUB 1502
1450 ? CHR$(125):POSITION 2,2:? " ":LI
ST A,B:POSITION 4,22:? "HIT <<RETURN>>
 AFTER ALL CHANGES"
1455 POSITION 4,23:? "ARE MADE"
1460 POSITION 2,1:INPUT CURSOR$
1480 POSITION 0,0:POKE 842,13:POSITION
 2,20:? "CONT":POSITION 2,0:STOP
1490 POKE 842,12
1492 ON COUNT GOTO 1508,1514,1520
1500 ? CHR$(125):GOTO 920
1502 IF NULINE<1080 THEN A=1020:B=NULI
NE
1504 IF NULINE>=1080 THEN A=1020:B=108
0:COUNT=1
1505 IF NULINE=1020 THEN COUNT=4:GOTO
920
1506 RETURN
1508 IF NULINE<1160 THEN A=1090:B=NULI
NE:COUNT=0
1510 IF NULINE>=1160 THEN A=1090:B=116
0:COUNT=2
1512 GOTO 1450
1514 IF NULINE<1230 THEN A=1170:B=NULI
NE:COUNT=0
1516 IF NULINE>=1230 THEN A=1170:B=123
0:COUNT=3
1518 GOTO 1450
1520 A=1240:B=NULINE:COUNT=0:GOTO 1450
1590 ? CHR$(125):N=1:POKE 709,10:POKE
752,1:TIME=1:LIMIT=1:IF NULINE=1320 TH
EN 920
1600 POKE 84,7:POKE 85,33:? "|":POKE 8
4,4:POKE 85,4:? "ENTER PROGRAM NAME ":
POKE 710,50:POKE 712,50
1602 FOR X=7 TO 12:POKE 84,X:POKE 85,3
3:? "|":NEXT X
1604 POKE 85,4:? "PROGRAM NAME LIMIT";
1606 FOR X=1 TO 11:? "▬";:NEXT X:? "▣"
1610 POSITION 4,2:? "ENTER <<DONE>> WH
EN FINISHED"
1619 IF T=1 THEN T=0:POKE 710,68:POKE
712,68
1620 POSITION 4,6:INPUT TP$
1630 IF TP$="DONE" THEN 1910
1640 TL=LEN(TP$):TIME=TIME+1
1650 IF TL>28 THEN GOSUB 2200:GOSUB 22
60:GOTO 1600
1655 IF ASC(TP$)>=48 AND ASC(TP$)<=57
THEN GOSUB 2200:GOSUB 2282:GOTO 1600
1660 IF TL<28 THEN TP$(TL+1)=BL$
1670 START=(N-1)*28+1
1680 A$(START)=TP$
1690 N=N+1:? CHR$(125):POKE 710,196:PO
KE 712,196
1700 POSITION 4,4:? "ENTER PROGRAM <<S
TART>> COUNT "
1710 TRAP 1759:POSITION 4,6:INPUT TP:T
P$=STR$(TP)
1720 IF TP>999 THEN GOTO 1759
1730 IF TRACE3=1 THEN TRACE3=0:IF TP<=
VAL(A$(START,START+27)) THEN 1790
1740 IF TRACE=1 THEN RETURN
1750 GOTO 1820
1759 GOSUB 2200:POKE 710,66:POKE 712,6
6
1760 ? CHR$(125):POSITION 2,11:? "PROG
RAM COUNT MUST BE NO MORE  THEN 3 DIGI
TS":TRACE1=1
1770 IF TRACE=1 THEN 1880
1780 GOTO 1700
1790 IF TRACE4=1 THEN TRACE4=0:? CHR$(
125)
1799 POKE 710,66:POKE 712,66:GOSUB 220
0
1800 ? CHR$(125):POSITION 4,10:? :? "P
ROGRAM END COUNT MUST BE LARGER THEN P
ROGRAM START COUNT ":TRACE4=1
1810 ? :? " START COUNT =  ";A$(START,
START+27):GOTO 1880
1820 TL=LEN(TP$)
1830 IF TL<28 THEN TP$(TL+1)=BL$
1840 START=(N-1)*28+1
1850 A$(START)=TP$
1860 N=N+1:? CHR$(125):POKE 710,210:PO
KE 712,210
1865 IF TRACE=1 THEN TRACE=0:LIMIT=LIM
IT+1:IF LIMIT=8 THEN 2300
1870 IF TRACE2=1 THEN TRACE2=0:GOTO 16
00
1880 POSITION 6,4:? "ENTER PROGRAM <<E
ND>> COUNT ":TRACE=1
1890 TRAP 1759:POSITION 6,6:INPUT TP:T
P$=STR$(TP)
1900 TRACE2=1:TRACE3=1:GOSUB 1720:GOTO
 1820
1910 IF TIME=1 THEN K=1:GOTO 2060
1915 LINENUM=NULINE:POKE 709,2
1920 ? CHR$(125):POSITION 2,2:? " "
1930 FOR X=1 TO N-1
1940 START=(X-1)*28+1
1950 IF ASC(A$(START,START+27))>=48 AN
D ASC(A$(START,START+27))<=57 THEN GOS
UB 2070:GOTO 2000
1960 ? LINENUM;" ";D$;" ";
1970 ? A$(START,START+27);
```

continued on page 115

OSCAR'S 4-WAY SPECIAL OFFER!



If you want to get more benefit out of your home computer and still avoid the purchase of expensive peripherals... if complex keyboard programming leaves you cold and you're tired of paying exhorbitant prices for every new cassette or diskette program—then you're ready for OSCAR and the DATABAR Club!

OSCAR is simply the finest hand-held optical barcode reader ever developed for the home computer marketplace. It works with virtually every home computer. And together with a simple plastic template, it allows you to enter all kinds of programs into your home computer in as little as two minutes.

But hardware without programs is useless—and that's where the DATABAR Club comes in. This includes a magazine that is delivered straight to your home every month with 8 programs on subjects ranging from fun and games, to education, to home finances. There are actually eight different categories in all: Funware, Homeware, Healthware, Wordware, Classware, Scienceware, Legalware, and Genware. Instead of paying $15 or $30 or more for cassette or diskette programs, or paying the $9.95 retail price for DATABAR programs—you can receive them for as little as $1.25 each by joining the DATABAR Club.

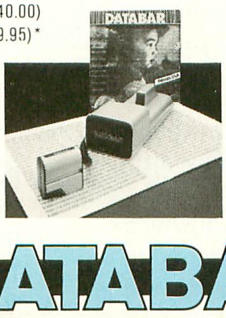
So if you already have a home computer, or if you're thinking about buying one—take advantage of OSCAR's 4-way offer now. Get your full money's worth on your computer investment. Send in the coupon, or call for further information.

Please send me the Databar option(s) indicated.
☐ Option 1 ($10.50) ☐ Option 3 ($240.00)
☐ Option 2 ($120.00) ☐ Option 4 ($79.95)*
*30-day money-back guarantee

Enclosed Check # ______

Money Order # ______
Payable to Databar.

Or charge my Am Ex ☐ Visa ☐
MasterCard ☐ Diners Club ☐
Carte Blanche ☐



Card # ______

Exp. Date ______

Name ______

Address ______

City ______ State ______ Zip ______

My Computer Model is ______

DATABAR

Databar Corporation, 10202 Crosstown Circle • Eden Prairie, MN 55344 • Phone Orders: MN (612) 944-5700
Out-of-State: 1 (800) 672-2776

PAGE FLIPPING

A racy tutorial

by DAVID PLOTKIN

In the August issue of ANTIC, the game "Maze Maniac" made use of one of the Atari's many special features, "page flipping." Page flipping is a powerful technique. It flashes a complete picture on the screen in an instant; the viewer does not have to wait while the picture is drawn.

In Maze Maniac, the player negotiates a maze shown on the screen, but each time he moves off the edge of the screen another maze appears instantly. While the accompanying article touched briefly on how this was done, the purpose of this article is to explain through example the mechanics of page flipping.

Why bother? The answer is speed. In the maze game, you change screens every few seconds. If you had to wait for each new screen to be drawn, you would soon tire of the game. With page flipping, a complete new screen is displayed in a split second. Full-screen animation is possible by making each picture just a little different from the one before. Flash them on the screen fast enough and the changes look continuous, like a movie. "Attack on the Death Star" (ANTIC, June 1982) works this way, and so does "Road Race," at the end of this article. If you plan carefully, your last frame can feed into the first, producing a continuous loop of animation. Full-screen animation is the most powerful use of page flipping.

Before we get into the nitty-gritty of page flipping, there are several important memory locations you should know about.

Memory locations 560 and 561 contain the address of the display lists in lo-byte/hi-byte form. This tells the Atari how to display a picture on the screen. To find the actual display list, use the equation:

```
DLIST = PEEK(560) + 256*PEEK(561)
```

Once you have found the display list, its fifth and sixth bytes (DLIST + 4 and DLIST + 5) contain the address of the chunk of memory to put on the screen. We will talk about modifying these memory locations later.

Memory locations 88 and 89 contain the address at which the Atari "thinks" the screen begins. This address is used in executing text and graphics commands to print and draw on the screen. While the contents of DLIST + 4 and DLIST + 5 are generally the same as the contents of 88 and 89 respectively, they do not have to be. You can change the values in 88 and 89 (write memory) from BASIC.

The concept of page flipping is really quite simple. First, you draw all your screen images during the initialization stage at various places in memory. To draw a picture other than on the screen, you merely adjust the values in locations 88 and 89 to point to the section of memory where you want the picture. Then execute your PLOTs and DRAWTOs. You won't actually see the picture being drawn, but it will be there in memory. When all the pictures are done and you are ready to use them, you must put them on the screen. This is done by adjusting the values in DLIST + 4 and DLIST + 5 to be the address where the picture is drawn (generally the same values as those you used in 88,89 when drawing the picture).

Let's take a short example. First, let's find a place to put our picture. We'll use the old standby, stepping back RAMTOP (location 106) to fool the computer into thinking we have less memory:

```
10 RT = PEEK(106)-12:POKE 106,RT:GRAPHICS
   0:DIM A$(1)
```

One "page" of memory equals 256 bytes. Since we have stepped back 12 pages, or 3K, and each GR.0 screen takes up about 1K, we now have room to draw three GR.0 pictures. We'll fill Screen 1 with asterisks:

```
20 POKE 88,0:POKE 89,RT:A$ = "*":GOSUB
   150
```

Note how we pointed 88 (lo-byte, which happens to be zero here) and 89 to the address of our first screen. Note also that the screen we are looking at didn't change while this was being executed. Now we'll fill Screens 2 and 3 with other characters:

```
30 POKE 88,0:POKE 89,RT + 4:A$ = "A":GOSUB
   150
40 POKE 88,0:POKE 89,RT + 8:A$ = "M":GOSUB
   150
```

To find the display list:

```
 50 DLIST=PEEK(560)+256*PEEK(561)
```

To print a message on the screen, we'll need to bring 88 and 89 back to point at the current screen you're looking at:

```
 60 POKE 88,PEEK(DLIST+4):POKE 89,PEEK
    (DLIST+5):POSITION 0,0:? "PRESS
    FIRE"
 70 IF STRIG(0)=1 THEN 70
 80 POKE DLIST+4,0:POKE DLIST+5,RT
 90 FOR REPEATS=100 TO 1 STEP-1
100 POKE DLIST+5,RT:FOR TIME=REPEATS TO
    0 STEP -1:NEXT TIME
110 POKE DLIST+5,RT+4:FOR TIME=REPEATS
    TO 0 STEP-1:NEXT TIME
120 POKE DLIST+5,RT+8:FOR TIME=REPEATS
    TO 0 STEP-1:NEXT TIME
130 NEXT REPEATS
140 GRAPHICS 0:STOP
```

The little routine from 80 to 130 flashes each picture on the screen successively, faster and faster, to show you how fast page flipping can be. Note that we only adjust the LMS hi-byte (DLIST+5) because the lo-byte has been set to zero (how convenient) for all of the screens. Finally, we have the small routine to put the characters on the screen:

```
150 FOR X=0 TO 38
160 FOR Y=0 TO 23
170 POSITION X,Y:PRINT A$;
180 NEXT Y:NEXT X:RETURN
```

As you can see, page flipping can move images around *fast*. There are some limits to page flipping, and we'll take those up now. First, the more pictures you store, the more memory must be reserved for the pictures. GR.0 and GR.5 screens each use about 1K of memory per picture. The redefined character sets use relatively little memory, yet give high resolution. Of course, you'll have to construct your pictures from redefined characters, and that can be tedious. GR.7 (3.5K per picture) and GR.8 (8K) require too much memory to be practical for page flipping.

Also, since you are looking at successive pages of memory, you really can't modify your animation much during a program. However, page flipping is ideal for making a moving background, relegating variable motion to Player/Missile graphics, which doesn't affect what is "underneath" it.

Finally, when page flipping is executed from BASIC, the illusion of speed can be lost because the speed isn't constant — the frequency of flipping (and thus its apparent speed) depends on how many other things are happening. For example, as the number of statements executed in each loop varies due to IF/THEN or FOR/NEXT statements, the frequency of page flipping also varies. This can be fixed by using *interrupts*, which free BASIC from the page flipping business altogether.

You've probably heard about vertical blank interrupts, and how you can put your own program into the VBI so that it will execute every sixtieth of a second, without affecting BASIC. The system timers work similarly, but are somewhat more flexible than VBIs. We are going to develop a page-flipping routine on the system-timer interrupt which will run fast and smooth, and which you can adjust from a running program.

To install your own machine-language routine in System Timer 2, which is available for your use, you need only place the address of your routine in the System Timer 2 vector: \$0228 (lo-byte, 552 decimal) and \$0229 (hi-byte, 553 decimal). Next, you place a value (in jiffies, or sixtieths of a second) into the countdown register, \$021A (decimal 538). Every sixtieth of a second, the countdown register is decremented. When it reaches zero, the Atari vectors to your routine and executes it. Your routine should finish by placing a new value in the countdown register. A zero will prevent your routine from being executed.

I have included the assembly-language routine for your study; you do *not* need to type it in to play Roadrace, as these instructions are already included in the BASIC listing DATA statements (lines 5060–5080). The advantage in using the countdown timer is that you can decide how often you want your routine to execute, rather than executed with *every* VBI, which is often too fast for page flipping. If you don't believe me, change SPD in line 5340 to:

SPD=100

Pretty fast, huh? The routine is executing with every VBI, and is a flickering blur. In the program, the name of the memory location that contains the value to be put in the countdown timer (1571) is SPEED. By adjusting the value of SPEED, you can make the roadway move at various constant speeds that are independent of BASIC.

To use this routine in your own programs, make the following adjustments:

1) Replace the 4 in line 5070 with the number of pages that each picture occupies. Note that this routine only adjusts the hi-byte, so each picture automatically starts on a page boundary.

2) Replace the 12 in line 5070 with the number of pictures multiplied by the number in (1) above.

3) Find the address of your first screen, as in line 5040, and insert it (hi-byte only) into the system-timer routine, as in line 5410.

The game I use to demonstrate page flipping is called Roadrace. To play, type it in (or take advantage of the copy offer) and run TYPO to eliminate the errors. Plug your joystick into Port 1 and RUN the program.

You control the race car at the bottom of the screen. Moving your stick left and right moves the car; pushing the stick forward increases your speed, while pulling it back slows you down. The object is to pass as many cars as possible. The score you get for passing each car depends on how fast you are going when you pass. Passing cars is difficult, because they

continued on page 97

PAGE FLIPPING continued from page 95

tend to weave back and forth across the two-lane highway. Your score, current speed and the time elapsed are shown at the bottom of the screen.

Roadrace is an endurance race. The time elapsed is shown in hours, and at the end of each 24-hour period you must have achieved a certain score or the game is over. As in real life, the game is also over if you crash into another car. Getting more than the necessary score allows you to race for another day. Extra touches, such as the sky darkening and the moon rising at night, lend added realism.

By the way, those of you who do not wish to type this program in may send me a disk (no cassettes, please) along with a stamped, self-addressed return mailer (or address label and postage, and I'll reuse your mailer) and $3.00 for a copy. Or, if you prefer, simply send $7.00 and I'll use my own disk and mailer. Send to:

David Plotkin
488 Walnut Ave.
Walnut Creek, CA 94598

No phone calls, please!

Listing 1

```
50      CDTMV2=$021A;THE COUNTDOWN FOR
TIMER #2
60      CDTMA2=0228;VECTOR FOR TIME #2
70      INDEX=$0630;1584
80        PTR=$0631;1585
90     DLBYTE=$0666;DUMMY
0100   *=$0600       ;START PAGE 6
0110   PLA           ;POP THE NUMBER OF A
RGS. OFF THE STACK
0120   LDA #$0B      ;LO BYTE OF TIMER RO
UTINE
0130   STA CDTMA2    ;PUT IN LO BYTE OF V
ECTOR
0140   LDA #$06      ;HI BYTE OF TIMER RO
UTINE
0150   STA CDTMA2+1;PUT IN HI BYTE OF V
ECTOR
0160   CLD           ;DO THIS OR BASIC WI
LL FOUL UP
0165                 ;YOUR ARITHMETIC
0170   LDA INDEX     ;KEEPS TRACK OF WHIC
H SCREEN YOUR ON
0180   CLC
0190   ADC #4        ;POINT TO NEXT SCREE
N (EACH SCREEN IS
0200                 ;4 PAGES LONG)
0210   CMP #12       ;TIME TO DISPLAY FIR
ST SCREEN AGAIN?
0220   BNE RESET     ;NO, GO DISPLAY 2ND
OR 3RD SCREEN
0230   LDA #0        ;YES, ZERO OUT INDEX
0240 RESET STA INDEX ;STORE IT
0250   LDA PTR       ;PTR CONTAINS HI BYT
E OF FIRST SCREEN ADDRESS
0260   CLC
0270   ADC INDEX     ;ACC NOW CONTAINS HI
 BYTE OF ADDRESS
0275                 ;OF NEXT SCREEN
0280   STA DLBYTE    ;DLBYTE WAS SET TO$0
666 IN THE EQUATES
0290                 ;BUT THE BASIC PROGR
AM REPLACES THE TWO
0300                 ;NUMBERS HERE WITH T
HE LO AND HI BYTES
0310                 ;OF THE DISPLAY LIST
 HI LMS BYTE ADDRESS
0320                 ;SO WE ARE STORING T
HE HI BYTE  OF THE NEXT
0330                 ;SCREEN INTO THE LMS
 HI BYTE,
0335                 ;THUS DISPLAYING IT!
0340   LDA #$02      ;RELOAD THE COUNTDOW
N TIMER
0350   STA CDTMV2
0360   RTS           ;ALL DONE
```

Listing 2

```
1 REM **** ROAD RACE *****
2 REM * BY DAVID PLOTKIN *
3 REM *  JANUARY, 1983   *
4 REM *  ANTIC MAGAZINE  *
5 REM ********************
10 GOSUB 5000:CAR1=0:CAR2=0:DIF=5:FLAG
1=0:FLAG2=0:TIME=0:POKE 19,0:DAY=0
100 SPD=SPD+10*(STICK(0)=14)*(SPD<90)-
10*(STICK(0)=13)*(SPD>50):POKE SPEED,1
1-SPD/10:SOUND 0,140-SPD,6,4
110 POKE 656,0:POKE 657,32:? SPD;:DIST
=(SPD-50)/10:IF SCR>1000 THEN DIF=1
120 IF CAR1<>0 THEN 170:REM MOVE CAR 1
130 IF RND(0)*DIF>1 THEN 200:REM SKIP
DOWN TO CAR 2
140 COL1=INT(RND(0)*16):POKE 706,COL1*
16+4:R=RND(0):CAR1=(R>=0.5)-(R<0.5)
150 C1X=104+14*(CAR1=1):C1Y=30
160 D=USR(ADR(JUMP$),PMB+768+C1Y,PMB+2
4):POKE 53250,C1X:GOTO 200
170 C1Y=C1Y+DIST:IF C1Y>84 THEN CAR1=0
:D=USR(ADR(JUMP$),PMB+768+C1Y-DIST,PMB
+64):SCR=SCR+SPD:FLAG1=0:GOTO 200
175 IF (RND(0)<0.1 AND C1Y>35) THEN CA
R1=-CAR1:FLAG1=1
180 IF FLAG1=0 THEN MX1=C1X+CAR1*(C1Y-
30)/2:GOTO 190
185 MX1=MX1+2*CAR1:IF (CAR1=1 AND MX1>
=118+(C1Y-30)/2) THEN FLAG1=0:C1X=118
186 IF (CAR1=-1 AND MX1<=104+CAR1*(C1Y
-30)/2) THEN FLAG1=0:C1X=104
190 D=USR(ADR(JUMP$),PMB+768+C1Y,PMB+2
4+12*(C1Y>50)+12*(C1Y>66)):POKE 53250,
MX1
```

continued on next page

TAKE-APART

10 Initialize variables, set up screens.

100 Read joystick and set flip speed and sound.

110 Print the speed, and decide how far each enemy car will move down the screen based on your speed. Reset the difficulty level at 1000 points.

120 Test to see if Car 1 is already on the road (CAR1<>0).

130 Car 1 is not on the road, so decide whether to put Car 1 on the road. Note that once your score exceeds 1000 points (DIF=1), you'll always put Car 1 on the road.

140 Choose a dark random color for Car 1, and decide which side of the road (lane) to start the car.

150 Set the X and Y coordinates (C1X and C1Y) based on line 140.

160 Place Car 1 on the screen and skip over the Car 1 move sequence.

170 Adjust the Y coordinate of Car 1 and test to see if it has gone off the bottom of the screen. If it has, erase it, bump up the score and turn off the "weave" flag (FLAG1).

175 Decide whether to weave Car 1 across the road. Reverse CAR1 to have the car change lanes, and turn on the weave flag.

180 We are not going to weave, so adjust the X coordinates based on the Y coordinates, so the car follows the edges of the road.

185 The car weaves. Adjust the X coordinate (MX1) to change lanes and test to see if you hit the opposite edge while moving right. If you did, stop the weave and adjust the base X coordinate (C1X) to be on the other side of the road.

186 Test to see if you hit the edge of the road while moving left. If you did, stop the weave and adjust the base X coordinate to the left side of the road.

190 Draw Car 1 on the screen. The third argument in the USR call decides what image to draw: the cars get larger and more detailed as they get closer (C1Y gets larger).

200–280 Essentially the same as 120–190 for Car 2.

285 Test to see if another hour of racing has elapsed.

290 Print the score and test for a collision.

300–390 Collision routine.

400 Test to see what time it is. If it's evening or dawn, turn the sky dark blue.

410 If it is late evening or early morning, turn the sky very dark blue.

420 Night. Turn the sky black and bring out the moon.

425 End of one day of racing. Make sky original color.

430 Print the time.

440 Test to see if another day has gone by.

450 Reset the timer to zero each day. If a sufficient score has been achieved, play for another day.

460 Erase all cars, stop page flipping and jump to "End of Game" routine.

5000 Step back RAMTOP to reserve memory for pictures. Set up area for cars (players).

5010 Clear memory by setting memory locations 88 and 89 to the beginning of the memory and doing a "clear screen" (CHR$(125)).

5020 Set RAMTOP and dimension D$ for picture drawing and JUMP$ to hold a machine-language subroutine.

5030 Find the display list, and the addresses of hi- and lo-bytes of LMS.

5040 Break the LMS hi-byte address into hi- and lo-bytes.

5050 Read in the machine code for the page flip routine.

5090 Set up the memory locations needed.

5108 The machine-language routine to instantly draw each screen from the data in lines 2000 to 20208. From ARTWORX's "DRAWPIC."

5120–5140 Redefine D$, set the area of memory to draw in (DAT) and execute the machine-language program from line 5108.

5150 Store the routine to move your car on the immediate VBI. The three-color car is two players overlaid for three colors.

5210 Store machine-language routine in JUMP$ to move the enemy cars on, around, and off the screen.

5240 Store the player shapes in the empty space at the beginning of P/M memory.

5300 Install immediate VBI player move routine.

5310–5320 Set colors, priorities, and widths.

5330 Put your car on the screen, set priority to get three colors (POKE 623,33).

5340 Modify the display list to customize the text window.

5350 Print the necessary quantities in the text window.

5360 Racing car sound.

5400 Initialize the INDEX and PTR memory locations.

5410 Stuff the lo- and hi-bytes of the LMS hi-byte address into the page-flipping routine.

5420 Point to the first picture and start the system timer.

```
200 IF CAR2<>0 THEN 260
210 IF RND(0)*DIF>1 THEN 290
220 COL2=INT(RND(0)*16):POKE 707,COL2*
16+4:R=RND(0):CAR2=(R>=0.5)-(R<0.5)
240 C2X=104+14*(CAR2=1):C2Y=30
250 D=USR(ADR(JUMP$),PMB+896+C2Y,PMB+2
4):POKE 53251,C2X:GOTO 290
260 C2Y=C2Y+DIST:IF C2Y>84 THEN CAR2=0
:D=USR(ADR(JUMP$),PMB+896+C2Y-DIST,PMB
+64):SCR=SCR+SPD:FLAG2=0:GOTO 290
265 IF (RND(0)<0.1 AND C2Y>35) THEN CA
R2=-CAR2:FLAG2=1
270 IF FLAG2=0 THEN MX2=C2X+CAR2*(C2Y-
30)/2:GOTO 280
275 MX2=MX2+2*CAR2:IF (CAR2=1 AND MX2>
=118+(C2Y-30)/2) THEN FLAG2=0:C2X=118
276 IF (CAR2=-1 AND MX2<=104+CAR2*(C2Y
-30)/2) THEN FLAG2=0:C2X=104
280 D=USR(ADR(JUMP$),PMB+896+C2Y,PMB+2
4+12*(C2Y>50)+12*(C2Y>66)):POKE 53251,
MX2
285 IF PEEK(19)=TIME+1 THEN GOSUB 400
290 POKE 656,0:POKE 657,22:? SCR;:IF P
EEK(53260)=2 THEN 100
300 D=USR(ADR(JUMP$),PMB+768+C1Y,PMB+6
4):D=USR(ADR(JUMP$),PMB+896+C2Y,PMB+64
):POKE SPEED,0
310 FOR LOOP=1 TO 200 STEP 4:POKE 704,
PEEK(53770):POKE 705,PEEK(53770)
320 POKE PMB+600+INT(RND(0)*8),PEEK(53
770):POKE PMB+728+INT(RND(0)*8),PEEK(5
3770)
330 SOUND 0,LOOP,8,8:SOUND 1,200-LOOP,
8,8:POKE 53257,RND(0)*4:POKE 53256,RND
(0)*4:NEXT LOOP
340 SOUND 0,0,0,0:SOUND 1,0,0,0:POKE 6
56,1:POKE 657,0:? "GAME OVER-PRESS fir
e"
350 IF STRIG(0)=1 THEN 350
360 POKE 656,1:POKE 657,0:? "
                              ";:SPD=60:SC
R=0:POKE SPEED,11-SPD/10:POKE 704,50
```

```
361 REM LINE 360 HAS 35 SPACES
365 POKE 53256,3:POKE 53257,3:POKE 710
,148:D=USR(ADR(JUMP$),PMB+384+15,PMB+6
4)
370 POKE 705,70:D=USR(ADR(JUMP$),PMB+5
12+84,PMB):D=USR(ADR(JUMP$),PMB+640+84
,PMB+12)
380 POKE 656,0:POKE 657,0:? " score
   speed";:POKE 656,0:POKE 657,22:? S
CR;"          ";:POKE 657,32:? SPD;
390 D=USR(1536):SOUND 0,140-SPD,6,4:CA
R1=0:CAR2=0:DIF=5:POKE 53278,0:FLAG1=0
:FLAG2=0:POKE 19,0:TIME=0:GOTO 100
400 TIME=PEEK(19):IF (TIME=8 OR TIME=2
2) THEN POKE 710,146
410 IF (TIME=12 OR TIME=20) THEN POKE
710,144:D=USR(ADR(JUMP$),PMB+384+15,PM
B+64)
420 IF TIME=14 THEN POKE 710,0:D=USR(A
DR(JUMP$),PMB+384+15,PMB+76)
425 IF TIME=24 THEN POKE 710,148
430 POKE 656,1:POKE 657,0:PRINT TIME;"
:00";
440 IF TIME<>25 THEN GOTO 490
450 TIME=0:POKE 19,0:POKE 656,1:POKE 6
57,0:PRINT "0:00  ";:DAY=DAY+1:IF SCR>
5000*DAY THEN 490
460 D=USR(ADR(JUMP$),PMB+768+C1Y,PMB+6
4):D=USR(ADR(JUMP$),PMB+896+C2Y,PMB+64
):POKE SPEED,0:POP :GOTO 340
490 RETURN
5000 RAMTOP=PEEK(106):SCRBASE=RAMTOP-1
2:PMBASE=SCRBASE-4
5010 GRAPHICS 2+16:POKE 88,0:POKE 89,P
MBASE:? #6;CHR$(125)
5020 POKE 106,PMBASE:GRAPHICS 5:DIM D$
(900),JUMP$(25)
5030 DL=PEEK(560)+256*PEEK(561):HIBYTE
=DL+5:LOBYTE=DL+4
5040 HI=INT(HIBYTE/256):LO=HIBYTE-(HI*
256)
5050 RESTORE 5060:FOR I=1536 TO 1578:R
EAD A:POKE I,A:NEXT I
5060 DATA 104,169,11,141,40,2,169
5070 DATA 6,141,41,2,216,173,48,6,24,1
05,4,201,12,208,2,169,0,141,48
5080 DATA 6,173,49,6,24,109,48,6,141,1
02,6,169,2,141,26,2,96
5090 PTR=1585:INDEX=1584:SPEED=1574:DL
BYTE=1571
5108 PICT=ADR("hh¬Lh¬Kh¬Nh¬Mhh¬0hh*hh¬
P$0◢1M┌K◢@▶Pw▀%KeP¬K♣▌fL▀%Me0¬M♣▌fNJP[
◆")
5120 GOSUB 20000:DAT=SCRBASE*256:D=USR
(PICT,DAT,ADR(D$),20,40,20)
5130 GOSUB 20100:DAT=(SCRBASE+4)*256:D
=USR(PICT,DAT,ADR(D$),20,40,20)
5140 GOSUB 20200:DAT=(SCRBASE+8)*256:D
=USR(PICT,DAT,ADR(D$),20,40,20)
5150 RESTORE 5160:FOR I=1649 TO 1727:R
EAD A:POKE I,A:NEXT I:REM VBI PLAYER M
OVEMENT
5160 DATA 104,173,34,2,141,190,6,173,3
5,2,141,191,6,169,6
5170 DATA 162,6,160,136,32,92,228,96,2
4,173,192,6
5180 DATA 141,0,208,141,1,208,173,124,
2,208,6,206,192,6,206,192,6,173,125,2
5190 DATA 208,6,238,192,6,238,192,6,17
3,192,6,201,66,176,5,169
5200 DATA 66,141,192,6,201,156,144,5,1
69,156,141,192,6,76,188,6
5210 FOR I=1 TO 25:READ A:JUMP$(I,I)=C
HR$(A):NEXT I:REM BLOCK MOVE
5220 DATA 104,104,133,204,104,133,203,
104,133,207,104,133,206,160,0
5230 DATA 177,206,145,203,200,192,12,2
08,247,96
5240 PMB=PMBASE*256:FOR I=PMB TO PMB+8
3:READ A:POKE I,A:NEXT I:REM PM DATA
5250 DATA 0,0,0,0,24,60,24,219,24,255,
24,195
5260 DATA 0,0,0,0,36,0,24,0,195,24,195
,0
5270 DATA 0,0,0,0,0,0,0,0,0,24,36,36
5280 DATA 0,0,0,0,0,0,24,60,24,90,126,
90
5290 DATA 0,0,0,0,24,60,24,219,219,255
,219,195
5295 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,24,60,90,255,189,90,36,24
5300 IF PEEK(547)<>6 THEN D=USR(1649):
POKE 1728,130
5310 POKE 704,50:POKE 705,70:POKE 706,
170:POKE 707,210:POKE 53256,3
5320 POKE 53257,3:POKE 53258,3:POKE 53
259,3:POKE 54279,PMBASE:POKE 53277,3:P
OKE 559,46
5330 D=USR(ADR(JUMP$),PMB+512+84,PMB):
D=USR(ADR(JUMP$),PMB+640+84,PMB+12):PO
KE 623,(33+16)
5340 A=DL+45:POKE A,71:POKE A+3,6:POKE
 A+4,6:POKE A+5,65:POKE A+6,PEEK(A+7):
POKE A+7,PEEK(A+8):SPD=60:SCR=0
5350 POKE 656,0:POKE 657,0:? " score
   speed";:POKE 656,0:POKE 657,22:?
SCR;:POKE 657,32:? SPD;
5360 SOUND 0,140-SPD,6,4
5400 POKE INDEX,0:POKE PTR,SCRBASE
5410 POKE DLBYTE,LO:POKE DLBYTE+1,HI:P
OKE SPEED,11-SPD/10
5420 POKE LOBYTE,0:POKE HIBYTE,SCRBASE
:D=USR(1536):FOR N=0 TO 3:POKE 53252+N
,64-2*N:NEXT N:POKE 711,14
5500 RETURN
20000 D$(1)="▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶◸W_
▶▶▶▶▶▶▶▶▶U_▶w▶▶▶▶UUW▶▶▶▶▶▶▶▶UU▶yU▶▶▶▶▶◸
UU_▶▶▶▶▶▶▶▶◸U▶e▬_▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶UUW▶"
20001 D$(101)="▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶◸UiU▶▶▶▶
▶▶▶▶▶▶▶▶▶▶▶▶uUUU▶UUUUUUUUTUUUUUUU]UU
UUUUUUeq5eUUUUUUuUUUUUUUUUe♥▪eUUUUUU"
```

continued on next page

```
20002 D$(199)="]UUUUUUUUUe▣▣eUUUUUWUU
UUUUUYU|▣▣▣U▣UUUUWUUUUUUUYU▣▣▣=U▣UUUUU
▣UUUUUUYV▣▣▣▣U▣UUUUU_UUUUUUYZ▣▣▣▣▣▣UU"
20003 D$(297)="UUUUUUUUUUYT♥♥♥♥U▣UUUUU
UUUUUUUYP♥♥♥♥▬▣UUUUUUUUU▣U_@♥♥♥♥▣UYU
UUUUUUU▣U▣♥♥♥♥♥♥▣UYUUUUUUUU▣U▣♥♥♥♥♥♥"
20004 D$(395)="UYUUUUUUUU▣V▣♥♥♥♥♥♥*UYU
UUUUUUU▣Z▣♥♥♥♥♥♥▣▣YUUUUUUU▣j♥♥♥♥♥♥▣%
YUUUUUUUU▣T♥♥♥♥♥♥♥♥UYUUUUUUUU▣P♥♥♥♥♥♥"
20005 D$(493)="♥♥▬YUUUUUUUU▣@♥♥♥♥♥♥♥♥▣
YUUUUUUUU▣♥♥♥♥♥♥♥♥♥♥▣UUZUV▣UW▣♥♥♥♥♥♥♥
♥♥▣UUZUV▣U_▣♥♥♥♥♥♥♥♥♥?▣UZUV▣U▣@♥♥♥♥♥"
20006 D$(591)="♥♥♥♥♥▣▣UZUV▣U*♥♥♥♥♥♥♥♥♥
♥♥▣)UZUV▣V(♥♥♥♥♥♥♥♥♥♥♥♥*UZUV▣Z▣♥♥♥♥♥♥♥
♥♥♥♥♥*▣ZUV▣j▣♥♥♥♥♥♥♥♥♥♥♥♥▣%ZUV▣U♥♥♥♥♥♥"
20007 D$(689)="♥♥♥♥♥♥♥♥▣UZUV▣T♥♥♥♥♥♥♥♥♥
♥♥♥♥♥♥UZUV▣P♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥▬ZUV▣@♥♥♥♥♥♥
♥♥♥♥♥♥♥♥♥▣ZUW▣♥♥♥♥♥♥♥♥▣@♥♥♥♥♥♥♥▣▣U_▣♥♥♥♥"
20008 D$(787)="♥♥♥▣@♥♥♥♥♥♥♥♥▣U":RETURN
:REM ROADWAY
20100 D$(1)="▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▣W_
▶▶▶▶▶▶▶▶▶▶U_▣W▶▶▶▶▶UW▶▶▶▶▶▶▶▶▶UUyU▶▶▶▶▣
UU_▶▶▶▶▶▶▶▶▶UUe▣_▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶UUW▣"
20101 D$(101)="▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▣UiU▶▶▶▶
▶▶▶▶▶▶▶▶▶▶▶▶▶UUUU▣UUUUUUUUUU▣▣UUUUUUU]UU
UUUUUUUU▣▣UUUUUUUuUUUUUUUUVUA▬VUUUUUU"
20102 D$(199)="]UUUUUUUUVV♥▣VUUUUUWUU
UUUUUUVT♥♥VUUUUUWUUUUUUUUVP♥♥▣UUUUUUU
▣UUUUUVUW@▣♥▣UVUUUUU_UUUUUVU_♥▣♥▣UVUU"
20103 D$(297)="UUUUUUUUUVUh♥♥♥♥%VUUUUU
UUUUUUVU▣♥♥♥♥)VUUUUUUUUUUUVU@♥♥♥♥▣VUUU
UUUUUUUUVU♥♥♥♥♥▣VUUUUUUUUUUUVT♥♥♥♥♥♥V"
20104 D$(395)="UUUUUUUUU▣UW▣♥♥♥♥♥♥?UUY
UUUUUU▣U_@♥♥▣♥♥♥▣UUYUUUUUU▣U▣♥♥♥♥♥♥▣u
UYUUUUUU▣U♥♥♥♥♥♥♥♥()UYUUUUUU▣V▣♥♥♥♥♥♥"
20105 D$(493)="♥♥*UYUUUUUU▣Z▣♥♥♥♥♥♥♥♥♥▣
▣YUUUUUU▣U♥♥♥♥♥♥♥♥♥♥▣UYUUUUUU▣T♥♥♥♥♥♥♥♥
♥♥UYUUUUUU▣P♥♥♥♥♥♥♥♥♥♥▬YUUUUUU▣@♥♥♥♥♥"
20106 D$(591)="♥♥♥♥♥▣YUUU▣UU▣♥♥♥♥♥♥▣♥♥♥
♥♥▣▣UUV▣UW▣♥♥♥♥♥▣♥♥♥♥♥♥▣UUV▣U_▣♥♥♥♥♥▣♥
♥♥♥♥♥?▣UV▣U▣@♥♥♥♥♥♥▣♥♥♥♥♥♥♥▣UV▣U*♥♥♥♥♥"
20107 D$(689)="♥♥♥♥♥♥♥ )UV▣V(♥♥♥♥♥♥♥♥
♥♥♥♥♥♥*UV▣Z▣♥♥♥♥♥♥♥♥♥♥♥♥♥♥*▣V▣j▣♥♥♥♥♥♥♥
♥♥♥♥♥♥♥♥▣%V▣U♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥▣UV▣T♥♥♥♥"
20108 D$(787)="♥♥♥♥♥♥♥♥♥♥♥♥♥UV":RETURN
:REM FIRST ROAD
20200 D$(1)="▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▣W_
▶▶▶▶▶▶▶▶▶▶U_▣W▶▶▶▶▶UW▶▶▶▶▶▶▶▶▶UUyU▶▶▶▶▣
UU_▶▶▶▶▶▶▶▶▶UUe▣_▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶UUW▣"
20201 D$(101)="▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▣UiU▶▶▶▶
▶▶▶▶▶▶▶▶▶▶▶▶▶UUUU▣UUUUUUUUUV\VUUUUUUU]UU
UUUUUUUV▣&UUUUUUUuUUUUUUUUUV@▣UUUUUUU"
20202 D$(199)="]UUUUUUUUeW▣▣UeUUUUUWUU
UUUUUUeh♥♥%eUUUUWUUUUUUUUe▣♥♥)eUUUUUU
▣UUUUUUUe@♥♥▣eUUUUUU_UUUUUUUe♥♥♥▣eUUU"
20203 D$(297)="UUUUUUUUUU▣U|♥♥♥▣UYUUUU
UUUUUU▣U▣♥♥♥=UYUUUUUUUUUUU▣Z▣♥♥♥♥▣▣YUU
UUUUUUUU▣j♥♥♥♥♥▣%YUUUUUUUUUUU▣(♥♥♥♥♥♥)"
20204 D$(395)="YUUUUUUUUUU▣P♥♥♥♥♥♥▬YUU
UUUUUUUU▣@♥♥♥♥♥♥▣YUUUUUUUUUU▣♥♥♥♥♥♥♥▣Y
UUUUUUU▣UU▣♥♥♥▣♥♥♥♥▣UUYUUUU▣UW▣♥♥♥♥▣♥♥"
20205 D$(493)="♥♥?UUYUUUU▣U_@♥♥♥♥▣♥♥♥♥▣
▣UYUUUU▣U*♥♥♥♥♥♥♥♥♥▣)UYUUUU▣V(♥♥♥♥♥♥♥♥
♥♥*UYUUUU▣Z▣♥♥♥♥♥♥♥♥♥♥*▣YUUUU▣j▣♥♥♥♥♥"
20206 D$(591)="♥♥♥♥♥▣%YUUUU▣U♥♥♥♥♥♥♥♥♥♥
♥♥▣UYUUUU▣T♥♥♥♥♥♥♥♥♥♥♥♥♥UYUUUU▣P♥♥♥♥♥♥♥
♥♥♥♥♥▬YUUUU▣@♥♥♥♥♥♥♥♥♥♥♥♥♥▣YUUUU▣♥♥♥♥♥♥"
20207 D$(689)="♥▣♥♥♥♥♥♥♥▣▣UUUW▣♥♥♥♥♥♥▣♥♥
♥♥♥♥♥♥▣UUU_▣♥♥♥♥♥♥♥▣♥♥♥♥♥♥♥?▣UU▣@♥♥♥♥♥♥
▣♥♥♥♥♥♥♥♥▣UUV*♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥▣*UZ(♥♥♥♥"
20208 D$(787)="♥♥♥♥♥♥♥♥♥♥♥♥♥♥*▣":RETURN
:REM SECOND ROAD
```


TYPO TABLE

Variable checksum = 1721727

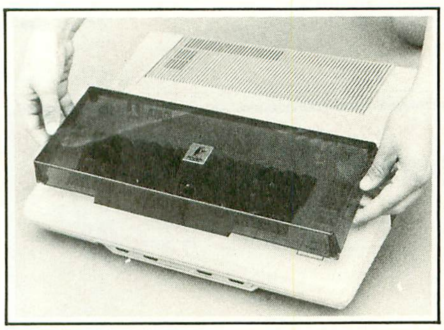
Line num range		Code	Length
1	120	SS	522
130	185	SZ	557
186	260	VH	527
265	300	LQ	550
310	360	WI	603
361	390	EL	527
400	460	WZ	539
490	5090	XX	524
5108	5180	CG	534
5190	5290	KF	524
5295	5340	OX	584
5350	20001	UI	581
20002	20006	KK	585
20007	20102	JX	516
20103	20107	NR	585
20108	20203	HI	519
20204	20208	ZJ	518

EAGLES
(game)
Strategic Simulations, Inc.
883 Stierlin Rd., Bldg. A-200
Mountain View, CA 94043
(415) 964-1353
48K — diskette
$39.95

Initiate missions, challenge duels, or instigate dogfights for up to twenty planes in this game that carries you through the skies of WWI France. The stable of flyers includes historical planes or those of your own design. Can be played against the computer or other players.

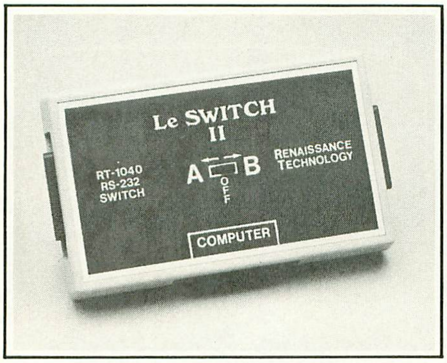
SMOOTHWRITER
(word processor)
Digital Deli
3258 Forest Gale Dr.
Forest Grove, OR 97116
(503) 627-9417
48K — diskette
$79.00 + $3.00 for shipping and handling

Intended for both the casual and the professional writer, this word processor is specially designed for use with Epson printers that have the Graftrax Plus feature. Its four programs are: text editor, file partitioner, text formatter and format reviewer.



K-COVER
(keyboard protector)
Penguin Products
P.O. Box 7008
Roseville, MI 48305
(800) 732-0614
$9.95

This durable shield prevents static, dust and spills from finding a home in your computer keyboard. It also doubles as an angled riser for computer or monitor, and comes with a lifetime guarantee against breakage.



LE SWITCH
(printer accessory)
Renaissance Technology Corp.
1070 Shary Circle
Concord, CA 94518
(415) 676-5757
Le Switch I (parallel) — $155.00
Le Switch II (serial) — $125.00

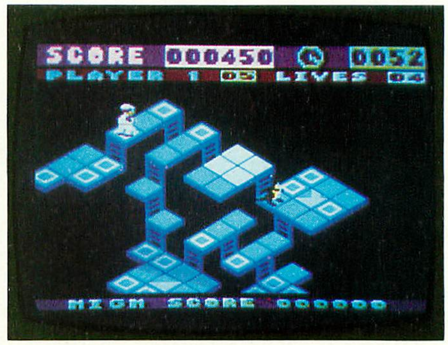
Perform word-processing and editing tasks on a high-speed, dot-matrix printer, then — voila! — switch over to a letter-quality printer for the final document. Two printers can run on a single computer with the aid of Le Switch, which is available for both serial and parallel printers.

PAPER TRACTOR
(printer accessory)
Paper Tractor Ltd.
One South Fairview, Unit C
Goleta, CA 93117
(805) 683-2851
$11.95

Easy to use and dependable, this plastic slipcover device facilitates the feeding of letter and legal size papers through any adjustable tractor-feed printer. Other items, such as checks and invoices, can also be tucked into it. No hardware modification is required.

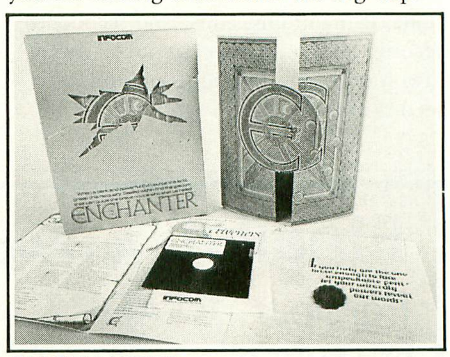
ACI-3 CASSETTE INTERFACE
(hardware)
Micro Systems and Software, Inc.
Contact Station #22
9297 Federal Blvd., Box 307
Denver, CO 80221
$44.95

In July, we published the wrong ordering address for this product, which interfaces any monaural or stereo cassette-tape recorder/player to the 400/800. Our apologies!



FLIP AND FLOP
(game)
First Star Software, Inc.
22 E. 41st St.
New York, NY 10017
(212) 532-4666
16K — cassette and diskette — $29.95
32K — cartridge — $39.95

Flip — you're a bouncing kangaroo! Flop — you're a swinging monkey! In this game, not only do you switch off between the marsupial and the primate modes, but the 3-D playfield also alternates between right-side-up and upside-down. If you overcome the scrolling platform maze, flying nets and an angry zookeeper, you hit the big time under the Big Top.



ENCHANTER
(game)
Infocom, Inc.
55 Wheeler St.
Cambridge, MA 02138
(617) 492-1031
32K — diskette
$49.95

The first in a new series of Infocom fantasy games, **Enchanter** includes the basic elements that the fledgling wizard requires: introductory spells, sealed parchment from the Circle of Enchanters, and an attractive Medieval Guild directory.

continued on next page

GAMES FOR YOUR ATARI COMPUTER

(book)
Dell Publishing Co., Inc.
1 Dag Hammarskjold Plaza
New York, NY 10017
(212) 605-3000
$5.95 U.S./$7.50 Canada

The Dell Computer Games series was an overnight success in England, where it was first published. The Atari version contains arcade-style games, brain teasers, puzzles, computer terms, bibliographies, and music — all in full color. Hail Brittania!

CHECKEASE!

(application)
T&F Software Co.
10902 Riverside Dr.
No. Hollywood, CA 91602
(213) 501-5845
16K — cassette — $29.95
16K — diskette — $34.95

With the advent of **Checkease!**, overdraft notices and bank statements may soon become archaic documents. This comprehensive checkbook-management system is similar to those used by banking institutions. It maintains, edits, and reconciles records of one or more accounts, and also functions as a budgeting and tax-record tool.



APROM

(data storage system)
Walling Co.
P.O. Box 2694
Cave Creek, AZ 85377
(602) 838-1277
cartridge — $99.95

APROM is a very sophisticated program development tool. A cartridge for the right slot of the Atari 800, it functions nominally as an EPROM programmer for 2764 (8K) and 27128 (16K) EPROMS. Extra functions include program controlled block switching and, with the APROM operating system, you can store various machine language programs in the APROM and execute them with a two-key command.



SHUFFLEBUFFER

(printing buffer)
Interactive Structures, Inc.
146 Montgomery Ave.
Bala Cynwyd, PA 19004
(215) 667-1713
32K — $299.00
64K — $349.00
128K — $445.00

Said to be the only printing buffer on the market with a random-access mode, **Shufflebuffer** enables the user to enter combinations of data, text, and graphics. Material also can be rearranged in any order for printing, plotting and telephone transmission.

Computer Outlet

COMPUTER OUTLET

We offer the largest selection of software and hardware for Apple, Atari, Commodore, IBM and Kaypro at 25 to 40% off retail.

1095 East Twain, LasVegas, NV 89109 • Mon.-Fri. 8AM to 6PM, Sat. 9AM to 5PM

600 XL $169
800 XL CALL

- 1010 Recorder $ 75
- 1050 Disk Drive $359
- 1027 Printer CALL
- 1025 Printer $459
- 830 Modem $165
- Touch Tablet $ 67
- Entertainer System $ 64
- 482 Educator $110
- 483 Programmer $ 52
- Communicator II $209
- Atari Accounting $169
- CX4104 Mailing List $ 19
- CXL4007 Music Composer $ 42
- Programming 2 & 3 ea. $ 22
- Conversational Languages ea. $ 45
- CX4018 Pilot $ 55
- CXB126 Microsoft Basic II $ 62
- CXL4022 Pac-Man $ 33
- RX8043 Ms. Pac-Man $ 39
- CXL4020 Centipede $ 33
- CXL4006 Super Breakout $ 26
- CXL4008 Space Invaders $ 26
- CXL4009 Computer Chess $ 26
- CXL4011 Star Raiders $ 30
- CXL4012 Missile Command $ 28
- CXL4013 Asteroids $ 28
- The Bookkeeper $102
- Home Filing Manager $ 38
- Atari Speed Reading $ 57
- Family Finance $ 38
- Time Wise $ 23
- Galaxian $ 33
- Defender $ 33
- Paint $ 33
- Qix $ 33
- Dig Dug $ 33
- ET Phone Home $ 34
- Atari Writer $ 75
- Donkey Kong $ 36
- Donkey Kong Jr. $ 38
- Atari Logo $ 79
- Football $ 36
- Pole Position $ 39
- Mickey in the Great Outdoors/D $ 34
- My First Alphabet/D $ 25
- Peter Pan's Daring Escape $ 34

Business & Utilities

- Visicalc $169
- Computari's Financial Wizard $ 45
- Color Accountant $ 65
- Mail Merge $ 20
- Data Perfect $ 95
- Letter Perfect $105
- Bank Street Writer $ 45
- Text Wizard $ 33
- File Manager 800 + $ 65
- Datasm 652.0 $ 59
- Paint Wizard $ 34
- K-Dos $ 59
- Lisp Interpreter $ 79
- Basic Compiler $ 55
- Datalink $ 27
- Atari World $ 39
- Color Print $ 27
- Graphics Generator $ 17
- Micropainter $ 23
- Graphics Master $ 17
- P.M.P. Property Management $149

Educational

Pre-School

- Sammy the Sea Serpent (C) $ 13 (D) $ 19
- Pre-School I.Q. Builder (C) $ 13 (D) $ 24
- My First Alphabet (D) $ 26
- Alien Counter/Face Flash (D, C) $ 26
- Humpty Dumpty/Jack & Jill (C) $ 25
- Facemaker (D) $ 23
- Juggles Rainbow (D, C) $ 22
- Juggles House $ 22
- Alphabet Zoo (CT) $ 20

Music

- Music Composer (CT) $ 42
- Jerry White Music Lessons (C) $ 20
- Magic Melody Box $ 14

Social Studies and Geography

- Globemaster (D) $ 27
- States and Capitals (C) $ 12
- European Countries and Capitals (C) $ 12

Math

- Monkey Up A Tree(D,C) $19
- Video Math Flash Cards(D,C) $13
- Algicalc(D,C) $19
- Compumath-Fractions (C)$23(D)$29
- Compumath-Decimals (C)$23(D)$29
- Golf Classic/Compubar (Angles) (D,C)$26
- Cash Register (C)$13(D)$19
- Big Math Attack (C)$17(D)$22
- Computation Concentration (C) $13 (D) $15

Reading and Language Arts

- Wordmaker (D, C) $ 19
- Spelling Genie (D, C) $ 19

- Compuread (C) $ 17 (D) $ 23
- Memory Builder/ Concentration (C) $ 13 (D) $ 19
- Let's Spell (C) $ 13
- Do-It-Yourself Spelling (C) $ 16
- S.A.T. College Board Prep (C) $ 89
- Vocabulary Builder I (C) $ 13 (D) $ 19
- Fishing for Homonyms (C) $ 13
- Hey Diddle Diddle (D) $ 20
- Snooper Troops 1 & 2 (D) Ea. $ 30
- Story Machine (D) $ 23
- Crossword Magic (D) $ 34
- Delta Drawing (CT) $ 23

Programming Techniques

- Pilot (Cons. of Educator) (C) $ 59 (D) $ 99
- Invitation to Prog. #2 (C) $ 22
- Invitation to Prog. #3 (C) $ 22
- Tricky Tutorials — Ed Software
  - TT#1 Display Lists (C, D) $ 17
  - TT#2 Horiz/Vert. Scrolling (C, D) $ 17
  - TT#3 Page Flipping (C, D) $ 17
  - TT#4 Basics of Animation (C, D) $ 17
  - TT#5 Player Missile Graphics (C, D) $ 24
  - TT#6 Sound & Music (C,D) $ 24
  - TT#7 Disk Utilities (D) $ 24
  - TT#8 Character Graphics $ 19
  - TT#9 GTIA $ 19
  - TT#10 Sound Effects $ 19
- Page 6 $ 20
- The Next Step $ 27

Stocking Stuffers

- **WICO "BOSS" Joystick $ 15.00**
- **WICO Redball Joystick $ 24.00**
- **Elephant Disks s/s $ 18.00**
- **Verbatim Disks s/d $ 24.00**
- **Disk Savers (Plastic Sleeves) Multi Colored 1 Doz. $ 4.50**
- **Disk Mailers $ 3.50**
- **Flip 'n File Diskette Holder w/Lock (holds 25) $ 18.50**
- **Flip 'n File Diskette Box (holds 50) $ 21.00**
- **Library Carrying Case (holds 10) $ 2.50**
- Koala Touch Tablet $ 69.00

Super Savers

- **Gorilla Banana Printer $ 209.00**
- **Percom Single Density Drive $ 389.00**
- **Percom Double Density Drive $ 515.00**
- **Amdek Color I Monitor $ 299.00**
- Rana 1000 Drive $299.00

Modems

- Hayes Smartmodem 300 Baud $209
- Hayes Smarmodem 1200 $499
- Signalman Modem II $ 79

Monitors

- Leading Edge Gorilla Hi-Res 12" $ 85
- Amdek Color I $ 299
- USI Amber $ 149
- NEC Green Screen $ 169

Printers

- Nec 8023A $ 475
- Epson FX80 $ 599
- Epson FX100FT $ 789
- Okidata ML82A $ 399
- Okidata ML83A $ 615
- Okidata ML84 P $1049
- Okidata ML92 P $ 469
- Okidata ML93 P $ 789
- Citoh 8510 Prowriter $ 365
- Mannesmann Tally 160L $ 629
- Qume Sprint 11 + $1429
- Gemini 10X $ 299
- Gemini 15 $ 479

New Hit List

- In Search of the Most Amazing Thing $ 27
- Witness $ 34
- Cosmic Balance II $ 27
- Temple of Apshai $ 27
- Raster Blaster $ 20
- Deadline $ 34
- Richochet $ 15
- Wiz & Princess $ 22
- Ali Baba and the Forty Thieves $ 22
- Canyon Climber $ 20
- Crush, Crumble & Chomp $ 20
- Zork I, II & III Ea.$ 27
- Frogger $ 23
- Choplifter $ 23
- Curse of Ra $ 15
- Ulysses and the Golden Fleece $ 23
- Tigers in the Snow $ 27
- David's Midnight Magic (D) $ 23
- Sky Blazer (D) $ 22
- Serpentine (D) $ 22
- Sea Fox (D) $ 20
- Sands of Egypt (D) $ 27
- Pool 400 (CT) $ 27
- Gorf (D)$ 27 (CT)$ 30
- Wizard of Wor (D)$ 27 (CT)$ 30
- Cyborg (D) $ 23
- Gold Rush (D) $ 23
- Bandits (D) $ 23
- Way Out (D) $ 27
- Fast Eddy (CT) $ 24
- World War (CT) $ 24
- The Cosmic Balance (D) $ 27
- Chess (D) $ 45
- Checker (D) $ 34
- Raptillian (D,C) $ 23
- Submarine Commander (CT) $ 34
- Jumbo Jet Pilot (CT) $ 34
- Soccer (CT) $ 34
- Starcross (D) $ 27
- Zaxxon (D,C) $ 27
- Miner 2049er (CT) $ 34
- Twerps (D) $ 23
- Flip Out (D) $ 20
- The Birth of the Phoenix $ 16
- Protector II (D)$ 23 (CT)$ 29
- Baseball (CT) $ 34
- Preppie II (D,C) $ 23
- Arcade Machine (D) $ 39
- Cap n' Cosmos (D) $ 29
- Spy's Demise (D) $ 15
- Repton (D) $ 27
- Critical Mass (D) $ 27
- Millionaire (D) $ 55
- Poker Sam (D) $ 17
- Jump Man (D) $ 27
- Hellfire Warrior (D,C) $ 27
- Trion (D) $ 27
- Adventure in Time (D) $ 20
- Wavy Navy (D) $ 23
- Final Orbit (CT) $ 23
- Pharoah's Curse (D,CT) $ 23

Order NOW For Christmas

To Order Call Toll Free 1-800-634-6766

Information & Inquiries 1-702-369-5523 • We accept VISA and MasterCard

ORDERING INFORMATION AND TERMS: *For Fast Delivery send cashier checks, money orders or direct bank wire transfers. Personal and company checks allow 3 weeks to clear. Charges for C.O.D. orders are $3.00 minimum or 1% for orders over $300. School purchase orders welcomed. Prices reflect a cash discount only and are subject to change without notice. Please enclose your phone number with any orders.* ***SHIPPING: — Software:*** *$3.00 minimum.* ***SHIPPING — Hardware:*** *(Please call)* ***SHIPPING — Foreign Orders:*** *APO & FPO orders: $10 minimum and 15% of all orders over $100. Nevada residents add 5 3/4% sales tax. All goods are new and include factory warranty. Due to our low prices, all sales are final. All returns must be accompanied by a return authorization number. Call 702-369-5523 to obtain one before returning goods for replacement.*

PUBLIC DOMAIN SOFTWARE

ANTIC GAMES DISK #1

1. Chicken: a great game from ANTIC V.1, No.1
2. Hangman: the traditional word game
3. Creation: a version of Life
4. Reverse: order of string numbers
5. Monopoly: computerized!
6. Lunar lander: select terrain (stick)
7. Zonex: hidden color patters in grid
8. Clewso: detective adventure, graphics

ANTIC GAMES DISK #2

1. Speed Demon: from ANTIC V.1, No.4
2. Guy' Grid Game
3. Deathstar: from ANTIC V.1, No.2
4. Blackjack: Vegas rules
5. Civil War: a strategic simulation
6. Artillery: firing strategy game
7. Super Wumpus: text adventure

ANTIC GAMES DISK #3

1. Petals
2. Showdown
3. FROG: from ANTIC V.1, No.3
4. Draw: Guy Hurt
5. Plus Zero
6. Collisi, and more

ANTIC GAMES DISK #4

1. Vultures: from Stan Ockers
2. Castle Hexagon: also by Stan Ockers
3. Adventure: The remainder of the disk contains an adventure game program which you can play, or you can use the program to write your own adventure. Instructions included in program.

ANTIC GAMES DISK #5

1. Bats: who else? Stan Ockers, from ANTIC V.1, No.5
2. Stellar Defense: from ANTIC V.1, No.6, slightly improved.
3. Yahtzee
4. Mastermind
5. Hamurabi: the classic simulation
6. Slalom: become a master schusser
7. Couch: analyze yourself
8. Aceyducy & more

ANTIC GRAPHICS DEMO #1

1. Spider: from ANTIC V.1, No.3
2. Rainbow
3. Horses
4. ATARI logo
5. Oxygen
6. Spiral
7. Pretty
8. Message and more

ANTIC PHOTO GRAPHICS

Digitized Photos

ANTIC UTILITY DISK #1

1. Doc: program allows you to accompany programs with separate documentation on disk.
2. Microassembler: allows you to create USR routines-assembler, more
3. Assembler-Editor: BASIC, slow but versatile
4. Num: automatic line numbering utility in BASIC
5. Memtest: runs without BASIC cartridge, to test all memory
6. Color: 128 colors at once
7. Printnop: connect parallel printer from jacks 3 & 4

ANTIC UTILITY DISK #2

1. Bubble Sort: From ANTIC V.1, No.4
2. Typo: from ANTIC V.1, No.3
3. Home inventory
4. KEY 6: Cipher coding
5. Renumber
6. Compare: listings for differences
7. SUPER: menu
8. Modem
9. RT clock & more

ANTIC UTILITY DISK #3

1. Disassembler: from ANTIC V.2, No.1
2. Tiny Text: from ANTIC V.1, No.6
3. GTIA text window: from ANTIC V.2, No.1
4. Label: disk label on Epson
5. Set up printer: sets up MX80 for Visicalc
6. Keyboard: tutorial & more

ANTIC GR. & SO. DEMO #1

1. Graphic
2. Draw
3. Rainbow
4. Tune Rite
5. Etch Sketch
6. Baby Pro Sound and more

ANTIC MUSIC DISK #1

Requires Music Composer Cartridge

1. Prelude
2. Joplin
3. In My Life
4. Star Trek
5. Daisy
6. Greensleeves
7. Yellow Submarine, and many more

Antic is pleased to offer a library of Public Domain Software for the Atari computers currently comprised of 12 disks. These programs are not yet available on cassettes. These disks contain unprotected material from the libraries of Atari users' groups from around the country.

The potential buyer should note that these programs are sold *as is*. Their usefulness may depend on your experience with the computer. They may contain programming quirks that require some modification. However, all perform reasonably well. Contents of the disks may vary slightly from the published description due to unforseen circumstances, but each disk is filled to reasonable capacity with useful programs of the kind described. In most cases, there is no documentation except that in the program themselves. An excellent value at $10.00 each, plus $1.50 per order for shipping/handling. Send check or money order (payable to ANTIC Publishing), and disk number(s) to:

Public Domain Software
524 Second St.
San Francisco, CA 94107

Allow four weeks for delivery. All orders are sent by First Class Mail. California residents, please add 6½% sales tax.

ANTIC presents these programs in diskette form for the convenience of the Atari community, in the belief that all of the programs offered are in the Public Domain and that no proprietary interests or rights to these programs are claimed by anyone. These diskettes are not copyprotected, nor does ANTIC claim rights to the programs themselves. The price of the diskettes is based on the cost of making them available.

TRACING NUMBERS

Flag that overflow!

by CLAYTON HOUSTON

Those of you who intend to do serious number crunching often need to trace calculations throughout your programs, usually for comparison with hand calculations of the problems. In Forth, this is difficult. There does not appear to be an easy way to print out calculations for validation. Further, if you have "overflow," you cannot even tell when and where it occurred. To improve this situation, I developed a couple of techniques that I will share with you here.

I am designing a game that requires speed, and I elected to program in Forth. The program contains more than 60 equations, most of which result in large numbers. As you know, there is a +/− 32768 maximum value in the integer, non-floating-point mode. Using existing Forth words, I scaled the calculations up and down — generally by 1000 — to keep the answers within the integer range. To accomplish this, I used a standard Forth definition, */, the notation for which is:

```
( n1 n2 n3 -- n )
```

This definition first uses the 32-bit register for the multiplication of the first two items, and then divides by the third item, leaving the answer as a single word on top of the stack. The value should be less than 32768.

In non-Forth notation, this operation is:

```
n = ( n1 * n2 ) / n3
```

In Forth, the data load-in might look like:

```
22222 3 100 */
```

which produces 666.

Well, I was moving right along, crunching through equations, when I discovered that random values exceeded the 16-bit limit of 32768, and overflowed, thereby losing the most significant bits of the answer. This was disastrous. To make matters worse, I couldn't tell where in the sequence the overflow(s) occurred. Though my knowledge of assembly language is sketchy, I was sure I needed to check the overflow bit of the Processor Status register.

Unfortunately, the PS register only shows overflow in addition, and I didn't know enough to break into the machine code so it would print a warning showing when and where an overflow occurred. I wasted a week of evenings thinking about ways to observe overflow.

Finally, there came a glimmer of hope. I was using */, which is a Forth word (definition) that I could modify. The definition for */ was essentially */MOD, which was:

```
: */MOD >R M* R> M/ ;
```

I decided to insert a DUP and PRINT after M* and M/, so that by comparing the two side-by-side values (which generally would be different by a scaling factor of 10, 100, 1000, etc.) I could see if the significant figures changed. If so, an overflow was likely. Examples of this would be:

66666	666	(okay, division by 100 probable)
22222	22222	(okay, no scaling)
66666	−1	(overflow, bad scaling chosen)

By tracing through and numbering each */ used on each screen in the program, I was able to identify exactly where overflow had occurred.

I also found a need to turn off the trace facilities, so I created a flag for this purpose. The calculations are in a big loop that prints out the final values once, at the end of the loop. I've injected the ?TERMINAL to break into the calculations at the end of the loop. When normal output shows that an overflow has occurred, I invoke the yellow terminal key, set the trace flag to one (1) with:

```
1 TFLAG !
```

and continue to run the program. Now the */ printout shows the suspect values.

To accomplish this, the following */MOD definitions are redefined:

```
0 VARIABLE TFLAG
: */MOD >R M* TFLAG @ IF
        2DUP D.
```

continued on next page

```
      R> M/ DUP . CR ELSE
      R> M/ ENDIF ;
: */ */MOD SWAP DROP ;
```

Note that the */ did not change from the original definition, but it must be redefined because the */MOD it calls is now a new definition.

Remember, you must first boot your Forth system, then load these new definitions, and then load your program in order for it to use the redefined sequence.

Still, I had occasional troubles. Because my number-crunching problem was so complex, I couldn't do a good hand calculation to check the answers. So I made a complete copy of my Forth disk and proceeded to convert the entire program to floating-point arithmetic. This was not as time consuming as I had feared; it was just a short-term fix to check the equations and their implementation. I still intended to return to integer calculations when I got the calculations correct. Having just conquered the */ integer problem, I was led to modify the floating-point operators in a similar way.

As you Forth users know, there are four main floating-point operators: F+, F–, F*, and F/. Again, the plan is to load the original operator definitions, then to insert the modified ones that suit your needs, and then to load the program. In this case, I wanted to see the result of each operation and to compare it with the hand calculations. The technique developed here is simply to perform a FDUP and then a F. each time an operation occurs. This prints out only the floating-point value, with no identification, so I inserted a ." to print the operator symbol as a clue to the origin of the number. The modified definitions become:

```
: F+ F+ FDUP CR ." + " F. ;
: F- F- FDUP CR ." - " F. ;
: F* F* FDUP CR ." * " F. ;
: F/ F/ FDUP CR ." / " F. ;
```

Then it was relatively easy to line up the symbol identification with each screen, and thereby have numerical values that could be compared directly for each operation to the hand-calculated values. This was progress! Of course, the TFLAG approach, used in the */ modification above, can also be used here to turn the trace on and off.

In my regular work as an engineer, I constantly use computer programs. To get them working properly, I usually trace and inspect the values of each step and compare them to hand calculations. Until now, I had not been able to do this at home. But using these little mods to the standard Forth operators, I am now able to follow the number crunching and have a way to find (most of) my errors. Maybe I'll soon get this monster program checked out and on the market for big bucks. If I don't, at least I will have had the satisfaction of discovering something new for this neat language called Forth.

POPEYE

Parker Brothers
Beverly, MA 01915
(617) 927-7600
$49.95, 16K — cartridge
Reviewed by David Duberman

Passion is no ordinary word, and **Popeye** is no ordinary video game. As we all know, Popeye loves Olive Oyl, but Brutus is determined to keep her for himself. This classic conflict of the cartoon series was masterfully retained in the arcade version. Popeye's coin-op incarnation was notable for being the first arcade game to feature an established cartoon character, as well as for its marvelous graphics.

The graphics have been simplified somewhat for the home version, but all the characters are still recognizable. When you turn on the game, you hear a brief rendition of Popeye's famous musical theme. Pressing [START] places you, as Popeye, on the highest of four platforms, with Brutus skulking about one level below. If you jump off this level, you can hit a punching bag and knock down a bucket. If your timing is just right, the bucket will land on Brutus' head and detain him for a few moments.

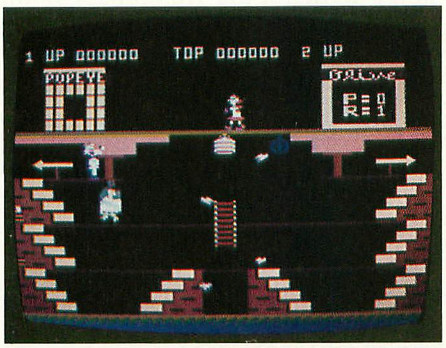
You also can move from level to level via stairs and a ladder. Meanwhile, Olive is trapped above, throwing pink hearts down to you as emblems of her love. You must collect 24 hearts in order to move to the next screen, or round. The sooner you catch each heart, the higher your score. During this interval, avoid Brutus. He constantly chases and tries to punch at you from the level above or below. He also tosses bottles at you, as does the Sea Hag. You can punch the bottles in self defense, but your timing must be impeccable.

Besides the bucket, you have (of course) some spinach with which you can temporarily banish Brutus. Use it wisely, however — one can is your quota. If you lose a life (you start with four), you get another bucket.

Round two also consists of a four-level house with stairs, ladders, and jumpways. The house is so colorful that it's sometimes hard to tell where the edges of the levels are. This time, Olive tosses 16 musical "love" notes from above.

The third screen places Popeye in his proper nautical environment, on a schooner. Olive now tosses letters: H-E-L-P. When all the letters are caught, you return to the first screen at a greater-difficulty level.

In sum, Popeye is a thoroughly entertaining challenge for gamers of all ages. It resembles other climbing games only in the employment of different levels; otherwise, it uses a unique play mechanic. It sustains a higher-than-average level of interest over a number of successive plays — a sure sign of good value.



AUSTIN-FRANKLIN 80-COLUMN BOARD

Austin-Franklin Associates
43 Grove St.
Ayer, MA 01432
(617) 772-0352
$289.95, 16K — hardware & cartridge
$69.95, RGB adapter

Reviewed by Mike Dunn

The **Austin-Franklin 80-Column Board** presents an 80-column by 25-character screen display on a monitor, using your Atari 800. It is the only 80-column board for the Atari that allows optional RGB (Red/Green/Blue) output. Although almost all monochrome monitors can display 80 columns, regular TV's and composite color monitors have only enough bandwidth for the usual Atari 40 columns. An RGB monitor is the only kind of color monitor that has the bandwidth to allow 80 columns.

This extra width is invaluable with some programs, especially word processors, and is very nice to have when programming with any language. With a word processor, you can easily preview the page before printing, and line up columns on the screen — tasks that are very difficult to do on a 40-column screen before printing on an 80-column piece of paper. If you have an SWP ATR-8000 with CP/M (see ANTIC, July 1983, page 109) or use a modem to access large computers, 80 columns is almost mandatory for serious work.

The board fits into the third slot of your Atari 800. In order to use the full 48K of memory in your Atari, it is necessary to use a 32K, 48K, or 64K memory board to free up the third slot. Early 32K boards tend to use too much power, making the video image unstable; Austin-Franklin recommends their 48K board with loop-back for optimal results. A ribbon is fed out of the Atari case, with one connection to your Atari monitor jack and the other to your monitor. If you use the RGB adapter, the monitor end of the cable is a standard 9-pin IBM PC-style monitor-output plug.

When using your computer in the regular 40-column mode, as with games, you will see the image on your monitor or TV as usual. Place a special cartridge in the right slot to convert the image to 80 columns — providing that certain requirements are met. Currently, most of the word processors designed for the Atari will not work, including those 80-column programs designed for the Bit3 80-column board. In fact, very few commercially-available programs will. The Austin-Franklin board does not require special software calls to work, but any program with "illegal" entry points

continued on next page

will crash. Although the right-cartridge-slot ROM does not use any memory, it will not work with a 16K left cartridge (e.g. AtariWriter). Both Austin-Franklin and LJK currently offer 80-column versions of Letter Perfect and Data Perfect that are compatible with the board. A forthcoming integrated series of applications programs from Synapse (Syn-Apps, including a word processor, data base, spreadsheet, etc.) will also be fully compatible.

Programs that will work with the board include the various languages available for the Atari, BASIC, BASIC A+, most assemblers, and some versions of Forth. Some terminal programs work very well, including the one that comes with the Microbits modem. Austin-Franklin makes a special ROM for the left slot ($30) for use with the ATR-8000 CP/M that is worth the price of the whole outfit alone, if you use CP/M.

Other features include a very nice 256 character set, as well as line and box graphics, and underline, blink, reverse and half-intensity capabilities. When used with the RGB monitor, you can shut off individual guns, so you can have a green monitor, for instance, or use all of the 16 colors. The RGB output will not work with Atari 40-column programs such as games, however — you need either a separate monitor or TV, or one of the few monitors that have both RGB and composite inputs for this! There is also a light-pen input; and software that emulates the DEC VT1000 terminal is being planned.

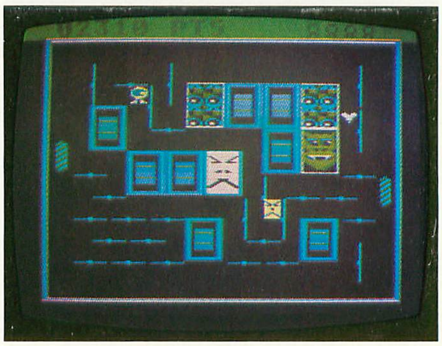
Is 80 columns worth all this money? The board is $280, and you also need a memory board (at least 32K), and a monitor ($100 and up for monochrome or $500 and up for RGB). If you have an ATR 8000 with CP/M, or want to access a large computer with a modem for serious use, I would say yes. If you do a lot of programming or word processing, 80 columns is very nice to have. In fact, I liked my review copy so much that I bought it!

DRELBS

Synapse Software
5221 Central Ave., #200
Richmond, CA 94804
(415) 527-7751
$34.95, 32K — disk & cassette
Reviewed by Bryan Welch

For many years, the Drelbs lived happily, without a worry in the world. Then the evil Trolloboars came, for they were a jealous race and wanted the Drelbish kingdom for their own. They brought with them the dreaded screwhead tanks and their sinister cousins, the Gorgolytes. Together, they conquered the kingdom and imprisoned all the poor Drelbs — except one.

As the last free Drelb, part of your mission is to defeat the Trolloboars on the



atomic flip grid. The grid consists of centrally-pivoted gates that rotate 90 degrees when you bump into them. To escape, you must form these gates into squares. This is done by flipping four of them around you, and thus creating a square.

But the Trolloboars and the tanks are trying to stop you. The Trolloboars chase you through the grid and mix up the gates. And if one of them touches you, it's bye-bye Drelb. The tanks, meanwhile, patrol the perimeter of the screen, firing deadly bullets that ricochet throughout the grid. Only when you complete all possible squares on the grid can you enter the Dark Corridor, and the second phase of the game.

In the Dark Corridor, you are confronted by a screen-full of imprisoned Drelbs and a Gorgolyte that shoots at you. You must free all the Drelbs (by walking over them) while evading the Gorgolyte's missiles. Should you succeed, you will have completed one of the eight rounds needed to win the game.

The sound and graphics in **Drelbs** are very well done. In the Dark Corridor, an ominous background rhythm becomes faster and more ominous as the game progresses. This makes the action very intense and may cause you to become nervous. The game's graphics are done in bright, cheerful colors, and fine scrolling effects are implemented in just the right places. Your Drelb watches where it is going, and even blinks!

Drelbs is an addicting game that you'll find very hard to put down, whether you are a beginner or a video-game whiz. Kelly Jones deserves a hand for an excellent job of programming. With great graphics and sound, and a neat idea, Drelbs is a real winner.

HUMPTY DUMP & FANCY WRITER

Kidstuff Software
3736 Ferndale
Ft. Wayne, IN 46815
(219) 485-2923
$34.95, 24K/48K — diskette

Reviewed by David Duberman

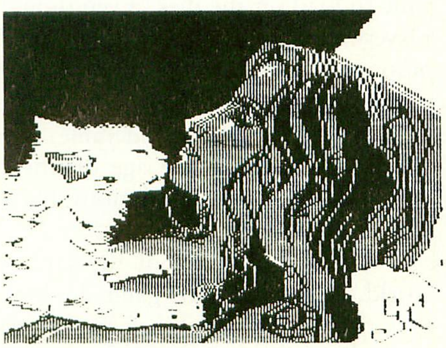
Lots of people are getting into computer art now. After all, it's a good feeling to be able to produce, by whatever means, an attractive picture on the monitor screen. This form of art has many advantages (e.g., no special materials are required, no mess is involved, and many special effects are available), but a primary disadvantage is that you can't hold the art in your hands when you're finished. The evanescent nature of the monitor screen, with its flickering image, imposes a transitory sense on all the images you create with it.

Humpty Dump is a RAM-resident, screen-dump program. This means that, once loaded, the program is available at

any time, even when another program is running. When you're ready to print what is on the screen, just press one console key and one keyboard key, and the image is transferred to your dot-matrix printer at machine-language speed.

Humpty Dump is designed to work with the following printers: Epson RX/FX with Graftrax, NEC 8023A, Prowriter, C. Itoh 8510A, and Gemini-10X. It may also work with other printers, if their graphics functions are similar to any of the above. For instance, we used the screen dump for the Epson on the Mannesmann Tally with good results.

With the Epson and Gemini printers, hard copy can be of two sizes, small or large. You can print the image as it appears on the screen (no color, of course),



or in inverse, as a negative image. With the others, you can also print a normal or inverse "super-duper size" picture that is about eight inches wide.

The Humpty Dump disk contains some extras, including three sample pictures. Also included are two BASIC listings. SAVEPIC.BAS is a subroutine that you can use with your own graphics program to save screens to disk. SHOWPIC.BAS is a program that lets you load and display previously-saved screens in any of the seven graphics modes, including the Micro-Painter mode. One interesting feature of this program lets you display the same picture in different graphics modes. Try displaying a Micro-Painter image in Graphics 10!

Fancy Writer, also from Kidstuff, allows you to print text in any one of five distinctive fonts. Each font ranges in size from tiny to large. It works with all of the printers mentioned above, plus the Axiom AT100 (Seikosha) printer. Fonts available are Cursive, Old English, Futura, Countdown, and Babyteeth. Four sizes of text are available.

This program is well thought out and quite complete. You can print from a pre-existing text file, or can use Fancy Writer to create one. You can also use the program as a typewriter, sending text to the printer directly from the keyboard. One final note: error trapping in both Fancy Writer and Humpty Dump is excellent.

Editor's Note: All screen dumps of "Janet" in this issue were made using Humpty Dump. Examples of fancy fonts were generated with Fancy Writer.



DEEP BLUE C & DEEP BLUE SECRETS

Atari Program Exchange (APX)
P.O. Box 3705
Santa Clara, CA 95055
(408) 727-5603
$39.95, 48K — diskette

Reviewed by Harald E. Striepe

The **Deep Blue C** compiler is the latest product of John H. Palevich, best known for "Chameleon," his excellent terminal emulator, also sold by APX. It is based on Ron Cain's public domain Small-C-Compiler. Since it's a proper subset of UNIX version 7 C, programs that do not use the special features of the Atari system can be recompiled on almost any other computer that supports the full language.

Deep Blue C is a pseudo-code (p-code) compiler. This gives it the advantage of smaller program size, but sacrifices speed. Still, considering the limited memory and disk capabilities of the Atari, this might be considered a valid compromise.

The Deep Blue C package is shipped with the compiler, the linker, and the object code for the p-code interpreter, along with quite-usable libraries for I/O routines (including the versatile printf (. . .)), graphics, and Player/Missiles. A couple of demonstration programs are also included.

The user will have to supply his or her own editor. The compiler does not accept line numbers, so you must use the "PRINT #D:. . ." command if you use the Assembler Editor cartridge. I also found MEDIT (APX) and Text Wizard by Datasoft to be useful tools. Since p-code is used, an assembler is optional, and is not required for high-level language use.

This implementation is reasonably complete, and I found it to be bug-free. The "for" and "switch" control structures make coding quite easy, as do "case" and "default." However, structures and unions, multi-dimension arrays, floating-point numbers, "sizeof" and type casting are not supported. Functions can return only integers. Assembly-language routines can be accessed using the "asm address" construct:

```
example(arg1,arg2)
  int arg1,arg2;
  asm 0x2300;
```

This calls an assembly routine at $2300. A usr(. . .) function similar to BASIC is also supported. This transfers arguments and their count on the machine stack. Even though this interface is not quite as elegant as that provided by a native-code compiler, it is still quite usable.

Modular compilation, together with the relocating linker, makes precompiled program libraries possible, as well as

continued on next page

providing an easy way to create larger programs. Since the compiler itself is written in C, it is quite slow, and thus the modular approach becomes essential.

The start of the resulting object files is at $3000. Thus, the system is compatible with all presently-existing operating systems for the Atari (KDOS has to be "squeezed," however). The manual is reasonably complete in describing the peculiarities of Deep Blue C and its libraries, but I highly recommend using it in conjunction with *The C Programming Language* by Kernighan and Ritchie (Prentice-Hall Software Series). A cross reference that outlines differences between the latter and Deep Blue C is thoughtfully provided in the package.

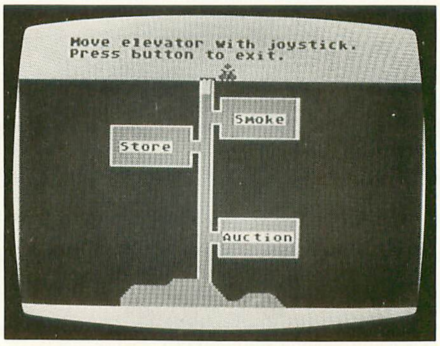
John H. Palevich is to be congratulated on convincing Atari to also publish the source code for Deep Blue C in a separate package, which is whimsically entitled **Deep Blue Secrets**. It contains the high-level source code for the compiler and the linker, as well as the assembly source code for the interpreter. This is written with Atari AMAC, but can be modified by the very ambitious to be used with other assemblers that support "include file" capabilities, such as the OSS MAC/65. The manual is concise. The moderately-experienced assembly programmer will get an opportunity to take a first look at the inner workings of a p-code engine, and will learn to modify the system to individual requirements.

In summary, the Deep Blue C package is a competent, fairly complete implementation. I strongly recommend purchasing the source package as well, if you decide to use this p-code system. I did not run specific bench marks, but it appears to be significantly faster than BASIC, and slower than both Forth and naturally, a native-code compiler like the OSS C65. On the plus side are the smaller code size and the price, since only $40 is enough to get you going (if you have some sort of text editor).

IN SEARCH OF THE MOST AMAZING THING

Spinnaker Software Corp.
215 First St.
Cambridge, MA 02142
(617) 868-4700
$39.95, 48K – diskette
Reviewed by Julie Sickert

In Search of the Most Amazing Thing, the latest game offered by Spinnaker Software, is everything good educational software should be – complex, challenging, and imaginative. The game is an adventure search that takes place on the planet Porquatz, a planet so large that much of it is unexplored. As the main character in the adventure, Terry



Bailey, you are searching for the Most Amazing Thing in the Entire Galaxy, something your old Uncle Smoke found at one time and then lost.

As you play, you will learn many of the skills and strategies necessary for making a long and eventful journey. For example, you'll find out how to equip the B-liner, a vehicle that is a cross between a tank and a hot-air balloon. You'll decide whether to drive or fly the B-liner, or to use your own jetpack. You'll learn to navigate by checking various meters for speed, height, fuel, and directions. You'll stop for food or sleep, or to trade with the Mire people, the only inhabitants of the Darksome Mire. You will find yourself plotting points on a map, calculating different rates of exchange, interpreting different languages and customs, keeping careful records of clues, and maneuvering to avoid the dangerous Mire Crabs.

This game is designed for one or two players using joysticks. A small book, enclosed with the diskette, details the history of Uncle Smoke Bailey's experiences in the Darksome Mire. A quick reference card and a booklet describing the format of the game complete the package.

Although In Search of the Most Amazing Thing is intended for ages ten through adult, I feel that it is more suitable for children of high-school age. It requires too much time and effort for the average ten-year-old child. The pacing of the text displayed on the screen cannot be controlled by the user, and the different routines in the program take some time to load off the disk. Even though players can SAVE a game in progress, the amount of persistence and organization necessary to advance through each stage of the adventure is considerable. Any child younger than the age of fifteen would have to be very well-motivated to persevere.

The multiple layers of the fantasy world presented in this game do, however, make good use of the best features of the Atari. The color, graphics, animation, and sound employed by the program are informative as well as captivating. The activities involved in playing the game are clever and stimulating. It is a great game for dedicated adventurers!

PAINT

Atari, Inc.
P.O. Box 61657
Sunnyvale, CA 94086
(408) 745-2000
(800) 538-8543 (outside CA)
$39.95, 48K – disk

Reviewed by Matt Shobe

Paint is an incredible graphics program for the Atari home computers. It is geared toward artistically-inclined computer users, but it can be used by just

about anybody who can doodle with a joystick. It can run on either an upgraded (to 48K) 400 or an 800, and on the new XL lines.

When Paint begins, you first see a neat title screen, then a menu with three choices: 1) simple Paint, 2) super Paint, or 3) Art Show. Simple Paint is designed for young children, or for days when you don't feel very creative or inspired. You choose from four brush sizes and paint with three basic colors. These are selected, via your joystick, from the 128 standard colors.

But Super Paint is where the fun really begins. This time you have four basic colors from which to choose, plus five "patterned" pots of paint. In addition, you can select almost any pattern imaginable from a color menu, and your four solid colors can be picked with a bit more precision. Also included is a *zoom* command that lets you paint pixel by pixel.

If you think this is enough, hold on — there's more. A feature called *draw* allows you to produce circles, lines and rectangles of any size and width automatically. You also have 81 different brushes at your disposal, and you can fill closed-in areas with any color or pattern using the *fill* command. Finally, Super Paint includes utilities that can load and save your masterpieces!

Art Show is simply a utility that displays your stored paintings. You select the sequence in which they are to be displayed and the length of time they will be shown.

Paint's manual is somewhat unusual. It's 175 pages long, but only 52 of its pages are devoted to instructions. The rest of the text explains how computers work, touches on the history of art and of computer art, and provides you with ideas for paintings.

My only criticisms of Paint are that it does not offer high-resolution graphics (GR. 7, 5 and 3 are used), and that there is no way to dump your graphics to a printer. A minor problem is the speed of the cursor — it may go too fast for you unless you set it to the lowest of its nine speeds. However, I recommend Paint to any ANTIC readers who like to draw with their computers, and to anyone who is skeptical about the graphics capabilities of the Atari. I really enjoyed using Paint, and I think you will too.

THE ALOG PAGEWRITER

Alog Computing
1040 Veronica Springs Road
Santa Barbara, CA 93105
(805) 964-4660 (CA)
(800) 558-8803 (outside CA)
$39.95, 32K — disk

Reviewed by Tay Vaughan

The real word is filled with compromises, and these are often dictated by the tools that are available to you. If you don't have a food processor, you use a knife to cut tomatoes. If yours isn't a sharp knife, you might have to quarter the tomatoes instead of thin-slicing them. In the worst-case compromises (with only a fork in hand), you might end up changing your menu to include freshly-squeezed tomato juice cocktail. The options and procedures available to you for cutting tomatoes, or for word processing on your Atari, are legion. In all cases, however, it helps to have the right tools for the job.

On your Atari, one of the major compromises in word processing is dictated by the computer's "standard" 40-column display. A letter or manuscript is usually printed in an 80-column format, but your screen can display only half of the full printed-line width (without special add-on ROM boards and hardware), so you cannot see a fully-composed page as you work.

The Alog Pagewriter provides an inexpensive solution to this composition problem and belongs in the category of special tools which perform specific jobs. It is not a full-blown word processor, and it intentionally compromises many fancy features to function as a simple, straightforward text manager.

Pagewriter orients the user in terms of the standard 8½ by 11-inch page, and its editing functions can be quickly mastered by users who are acquainted with the Atari keyboard or who have had experience with standard typewriters. Most of the screen is devoted to a compositional display of the page being written, including margins, line spacing, line index, and indents. This display, however, does not include written text and characters, but graphic blocks and spaces representing the way the page will look when it is printed. In another window at the bottom of the page, the actual text of the line being written or edited is displayed. If the text is longer than 40 characters or exceeds the chosen margin settings, the text line wraps around. Lines can be deleted, blank lines can be inserted, and individual words can be changed or corrected. You can view only one line of text at a time in the lower window, however, so reading what you have typed means scrolling line by line through your text. This makes it difficult to maintain continuity of thought when you are doing creative writing. For this, you may prefer a more expensive and capable word processor.

But for straightforward page-oriented writing tasks, Pagewriter performs well. Up to ten pages can be stored to disk for later recall, and control characters can be imbedded in the text for printer features such as condensed or elongated text. The system of storing page files to disk is not compatible with Atari DOS, however, so separately-formatted disks are required for each writing project.

As a simple word-processing tool, Pagewriter is useful in situations that require display of the finished composition and layout of the text. If composition is important to your work, this may be the special tool you need to avoid compromise.

continued on next page

HEXMASTER

Dolphin Microware
410 Stanford Ave.
Palo Alto, CA 94306
$19.95, 16K — cassette
$19.95, 32K — disk
Reviewed by Murray Kucherawy

In 1942, physicists at the Neils Bohr Institute in Copenhagen discovered a pleasant diversion from the tedium of soiving the mysteries of the atom. The diversion was the game "Hex," invented by Danish philosopher Piet Hein, and the scientists were intrigued by the game that takes "moments to learn, but a lifetime to master."

David Lischinsky has done a fine job of converting Hex for use on the Atari; the result is **Hexmaster**.

The game involves two players (or one player and the computer) who take turns occupying hexagons on the board. One tries to connect the top of the playfield to the bottom, while the other connects the left side to the right. As you attempt this, of course, you try to block your opponent. A tie is imposible. Sounds simple, doesn't it?

I currently own a 37-game losing streak against my Atari (at the lowest level of play), and my computer doesn't even seem to be apprehensive when it studies my moves.

Hexmaster's graphics are pleasing, but the program makes little use of the Atari's sound capabilities. You make your moves with a joystick and its fire button, and you have the ability to "take back" or restore moves — or even to replay the entire game. Throughout, the computer's responses to your moves are quite prompt.

The game is enhanced by three two-player options:

- Two-player timed option — In this version, each player's moves are timed. Thus, you can lose as the result of running out of time regardless of your progress. The times on each player's clock are selectable and allow for handicaps.

- Two-player speed option — Same as above, except that timers are not used, and you don't have to wait your turn. The result: Chaos!

- Two-player blitz option — Same as above, but faster. Insanity!

Hexmaster also features a bonus utility. Its digital game timers can be accessed independently for use in playing chess, Scrabble, go, and other timed games.

If you're tired of shoot-em-up games, and want to quietly match wits with your Atari or have some frantic fun with another human (of any age), Hexmaster is a worthy addition to your software library.

DANCING FEATS

Softsync, Inc.
14 East 34th St.
New York, NY 10016
(212) 685-2080
$24.95, 32K — cassette
$29.95, 32K — diskette

Reviewed by George Adamson

Have you ever seen the "Dancing Waters" fountains on TV or at a water park? If so, you will appreciate the novelty of **Dancing Feats**.

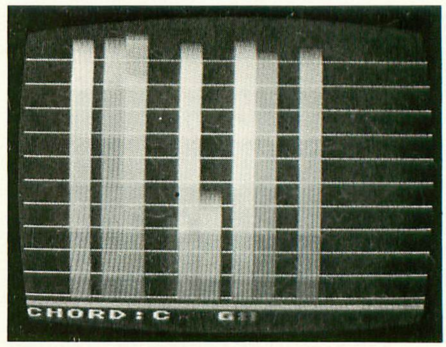
This welcome program turns your computer into a "one-man joystick band," while creating a dazzling display of rainbow jet sprays on the screen. Using a joystick, you select your bass line, beat, style, tempo, and type of ending from menus offering four choices for each musical component. You also use a joystick to select the actual notes. The only times you need to use the keyboard are to save and load files (to and from memory, disk or cassette), and once to end your composition.

Pressing the joystick produces note values displayed in the text window, rising and falling musical sounds, and multi-shaded geysers simultaneously. Pressing the fire button creates a note one octave lower. By firing rapidly, you can play notes from both octaves at the same time. Rotate the joystick in a circular motion, and you send a dazzling display of dancing colors across the screen.

The computer automatically selects the chords (C and F), but you can play different notes by pointing the joystick in different directions. Sometimes selecting notes accurately with a joystick is a challenge. Fortunately, the computer stays in tune no matter how you play. It is difficult not to be melodious.

The instruction sheet proclaims, "You don't have to be a musician to play." This reviewer can't read a note of music, but if I can enjoy Dancing Feats, anyone can.

Softsync gives due credit to the author, Christopher Chance, an accomplished



jazz, rock, and recording artist.

My son, who has studied the guitar, offered this comment: "The rainbow colors synchronized with the synthesized sound make you want to experiment, combining different notes into composition. You won't try it once and then forget it."

If you want to be the leader of the band, but are too shy or can't play an instrument, Dancing Feats may be the answer to your fantasy.

PRINTER SURVEY continued from page 56

resumes at the same place when paper is loaded. The ribbon for the Gemini is standard, old-fashioned, fabric typewriter ribbon on a spool. Considering the cost of ribbons for the highly-individualized machines that require cartridges, this is a real selling point.



THE SPIRIT

Mannesmann Tally
8301 S. 180th St.
Kent, WA 98232
(800) 447-4700

Mannesmann Tally is a name better known in business than in home computing. A European conglomerate, the company offers a range of printers for business use; the Spirit is at the low end of this range. You might call it a "professional" printer. Its sleek design, sound-muffling lid, and crisp font all suggest it for office use. Still, it is inexpensive enough for home consideration as well.

The Spirit is a dot-matrix printer with nine pins in the print head — but the pins are square rather than round. This gives an authority to the printed letters that requires no apology, even when used in business correspondence. A carbon-film ribbon is available for an even sharper image. The impact of the pins is variable and can produce up to three carbon copies plus an original. Its paper-handling mechanism, like the NEC's, feeds forms and stationery through tractors at the rear. As a result, a new sheet is always cued without waste.

The Spirit's character set of 228 symbols includes standard letters and italics, plus a selection of graphic elements that look to be useful for constructing charts and graphs. The Spirit is rated at 80 characters per second, and tested at 53 CPS in our benchtest. The Spirit responds as if it were an Epson when being driven by graphics programs. An audible alarm notifies the operator when the printer is out of paper.

All in all, The Spirit is a classy and very business-like printer.



NEC PC-8023A

NEC Home Electronics, Inc.
1401 Estes Ave.
Elk Grove Village, IL 60007
(800) 323-1728

At almost 20 pounds, this popular dot-matrix printer is a heavyweight. It's also a heavy-duty machine, and the head of a family of printers — so far as control codes are concerned — that includes the C. Itoh Prowriter. This family difference distinguishes it from the Epson and the Centronics types. As a result, a NEC owner has to hunt a little harder for the utility programs, such as screen dumps, that make graphics-capable printers so attractive.

Otherwise, the NEC is a very desirable printer. It is a well-designed, well-crafted, versatile and rugged workhorse. It accepts all kinds of paper, including carbon sets and rolls, and tractor-fed materials as narrow as 4½″. The tractors are in the rear, so paper waste is minimized and a new sheet is always cued for printing.

Many fonts are available, including Japanese "Katakana" symbols; selected characters from the Greek, German and Swedish alphabets; and an integral graphics character set that can be used to build images. Letter fonts can be printed in double-width, pica, elite, condensed and proportional faces. The NEC can underline and boldface, but has no subscripts or superscripts — a strange deficiency in such a capable printer.

The NEC can print either bidirectionally or one way, and can detect short lines. Rated at 120 characters per second, the NEC produced only 61 CPS on our benchtest, a bit on the slow side. The graphics capabilities of the NEC are good, but it is not as well supported by graphics utility programs as is the Epson. The PC-8023 has a 2K buffer, about enough to hold a full page of single-spaced characters.

Two-toned styling and angular shapes distinguish the NEC's appearance, and its controls and signal lamps are conveniently located. It occupies a bit less space than the Epson, which can be an important difference on a crowded computer table. The manual is slim, but comprehensive; unfortunately, it seems to have been written for technicians rather than home computerists.

This NEC is a little cheaper than the Epson FX-80, with which it competes most directly. The NEC seems to be built a little better, but it performs a little worse. The choice here is very much like that between cars — which do you like, a Chevy or a Ford?



TRANSTAR 120

Transtar
P.O. Box C — 96975
Bellevue, WA 98009
(206) 454-9250

Many documents are reputed to require characters that look as if they were made by a typewriter. Business and professional correspondence, reports, theses and manuscripts are often cited as examples. This appearance is called "letter quality," and the computer industry has devised several ways to achieve it.

The Transtar 120 is a letter-quality printer that uses daisywheel technology, and it is the only such printer in our survey. Most daisywheel printers cost much more than the Transtar, so we are pleased to present it as an economical option for our readers.

If you get a daisywheel printer, you forfeit the graphics capability that many hobbyists find fascinating. You also forfeit speed. The Transtar 120 is rated at

continued on next page

14 characters per second and we tested it at 12 CPS. That's only about one fifth as fast as a dot-matrix printer of equal cost.

The Transtar 120 can print in 10-pitch (pica), 12-pitch (elite), or 15-pitch (condensed). A different wheel is required for each pitch and font, but only one wheel is supplied with the printer. Extra wheels cost about $20 each. Each wheel has 96 letters or symbols in the desired font, such as italics. You must change wheels to change fonts.

Paper is held by a friction roller and can advance in increments of 1/48". The platen is 13 inches long, so standard paper can be inserted sideways, which is sometimes convenient. Subscripts and superscripts can be printed, and both underlining and boldface (overstrike) are available. Ribbon cartridges, fabric or carbonized mylar, are the same as those used in some Olivetti machines. If the ribbon jams or runs out, the Transtar stops operating and a red light warns you of the problem.

An autoload feature brings each sheet of paper to a preset point, and an eject button quickly rolls the sheet out when printing is over.

WE'RE MOVING

As of January 1, 1984 ANTIC'S new address will be

ANTIC PUBLISHING
524 Second Street
San Francisco, Ca. 94107

CALLING ALL TAPES continued from page 92

```
1980 LINENUM=LINENUM+10
1985 IF LINENUM>=1320 THEN 2015
1990 NULINE=LINENUM
2000 NEXT X
2005 FOR X=2520 TO 2524 STEP 2:? X:NEX
T X:? "2005 ? 945:? 1505 :? 2005"
2015 POSITION 2,22:? "545 ";"NULINE =
";LINENUM
2020 POSITION 0,0:? " ":POKE 842,13:PO
SITION 2,23:? "CONT":POSITION 2,0:STOP
2050 POKE 842,12
2055 IF LIMIT=8 THEN 1590
2060 ? CHR$(125):POKE 709,0:POKE 710,2
46:POKE 712,246:GOTO 920
2070 TL$=A$(START,START+27)
2080 IF TRACE6=1 THEN TRACE6=0:? TL$;"
)":GOTO 2100
2090 ? "(";TL$;"-";:TRACE6=1
2100 RETURN
2200 FOR L1=14 TO 2 STEP -0.5
2210 FOR L2=50 TO 75 STEP 20
2220 SOUND 0,75-L2,10,L1
2230 NEXT L2
2240 NEXT L1
2250 RETURN
2260 ? CHR$(125):POSITION 2,10:? "PROG
RAM NAME IS ";TL-28;" CHARACTERS TOO L
ONG":T=1
2270 POKE 710,68:POKE 712,68
2280 RETURN
2282 ? CHR$(125):POSITION 2,10:? "PROG
RAM NAME MUST NOT START      WITH A NU
MERAL"
2284 POKE 84,15:? "PROGRAM NAME ";TP$:
POKE 710,68:POKE 712,68:RETURN
2300 POKE 710,50:POKE 712,50
2302 ? CHR$(125):POKE 84,8:? :POKE 85,
14:? "LOADING DATA.":? :? :POKE 85,7:?
 "WILL CONTINUE IN A MINUITE"
2305 TRACE2=0:TRACE3=0:TRACE=0
2310 FOR DELAY=1 TO 250:NEXT DELAY:GOT
O 1915
2400 ? CHR$(125):POKE 84,14:? "NAME IS
 ";(LEN(Y$)-20);" TOO LONG"
2410 POKE 84,8:? :? "ENTER TAPE NAME "
:? :? :INPUT Y$
2420 IF LEN(Y$)>20 THEN 2400
2430 GOTO 529
2500 ? CHR$(125):POSITION 2,2:? " ":FO
R Y=521 TO 529:? Y:NEXT Y:FOR Y=2400 T
O 2430 STEP 10:? Y:NEXT Y
2502 ? "535 Y$ = ";CHR$(34):Y$;CHR$(34
)
2504 ? "530 X$ = ";CHR$(34):X$;CHR$(34
)
2506 POSITION 0,0:POKE 842,13:POSITION
 2,23:? "CONT":POSITION 2,0:STOP
2508 POKE 842,12
2510 FOR DELAY=1 TO 50:NEXT DELAY
2512 ? CHR$(125):POSITION 2,2:? " ":FO
R Y=2500 TO 2514 STEP 2:? Y:NEXT Y:POS
ITION 0,0:POKE 842,13
2514 POSITION 2,23:? "CONT":POSITION 2
,0:STOP
2520 POKE 842,12
2522 POKE 842,12
2524 GOTO 540
```

TYPO TABLE

Variable checksum = 1284882

Line num range			Code	Length
485	–	522	FK	500
523	–	600	DY	434
610	–	720	OS	433
730	–	830	XM	520
840	–	945	IG	453
950	–	1360	DM	470
1365	–	1412	HX	581
1420	–	1460	RY	507
1480	–	1514	JP	446
1516	–	1606	TM	504
1610	–	1700	OJ	497
1710	–	1800	VH	538
1810	–	1910	VV	501
1915	–	2005	SF	368
2015	–	2220	VL	458
2230	–	2305	VK	537
2310	–	2510	XK	526
2512	–	2524	UC	218

LISTING CONVENTIONS

Table Information

Our custom font listings represent each ATASCII character as it appears on the video screen. You generate some characters by a single keystroke, for example, the regular alphabet. Others require a combination or sequence of keystrokes. In this table, ESC means *press and release* the escape key before pressing another key. CTRL or SHIFT means *press and hold* the control or shift key while simultaneously pressing the following key.

The Atari logo key ([logo]) "toggles" inverse video for all alphanumeric and punctuation characters. Press the logo key once to turn it on; press again to turn it off. In the XL line there is no logo key; inverse video is controlled by a key on the function row. Decimal values are given as reference, and correspond to the CHR$ values often used in BASIC listings.

NORMAL VIDEO

FOR THIS	TYPE THIS	DECIMAL VALUE
	CTRL ,	0
	CTRL A	1
	CTRL B	2
	CTRL C	3
	CTRL D	4
	CTRL E	5
	CTRL F	6
	CTRL G	7
	CTRL H	8
	CTRL I	9
	CTRL J	10
	CTRL K	11
	CTRL L	12
	CTRL M	13
	CTRL N	14
	CTRL O	15
	CTRL P	16
	CTRL Q	17
	CTRL R	18
	CTRL S	19
	CTRL T	20
	CTRL U	21
	CTRL V	22
	CTRL W	23
	CTRL X	24
	CTRL Y	25
	CTRL Z	26
	ESC ESC	27
	ESC CTRL −	28
	ESC CTRL =	29
	ESC CTRL +	30
	ESC CTRL *	31
	CTRL .	96
	CTRL ;	123
|	SHIFT =	124
	ESC SHIFT CLEAR	125
	ESC DELETE	126
	ESC TAB	127

INVERSE VIDEO

FOR THIS	TYPE THIS	DECIMAL VALUE
	[logo] CTRL ,	128
	[logo] CTRL A	129
	[logo] CTRL B	130
	[logo] CTRL C	131
	[logo] CTRL D	132
	[logo] CTRL E	133
	[logo] CTRL F	134
	[logo] CTRL G	135
	[logo] CTRL H	136
	[logo] CTRL I	137
	[logo] CTRL J	138
	[logo] CTRL K	139
	[logo] CTRL L	140
	[logo] CTRL M	141
	[logo] CTRL N	142
	[logo] CTRL O	143
	[logo] CTRL P	144
	[logo] CTRL Q	145
	[logo] CTRL R	146
	[logo] CTRL S	147
	[logo] CTRL T	148
	[logo] CTRL U	149
	[logo] CTRL V	150
	[logo] CTRL W	151
	[logo] CTRL X	152
	[logo] CTRL Y	153
	[logo] CTRL Z	154
	ESC SHIFT DELETE	156
	ESC SHIFT INSERT	157
	ESC CTRL TAB	158
	ESC SHIFT TAB	159
	[logo] CTRL .	224
	[logo] CTRL ;	251
	[logo] SHIFT =	252
	ESC CTRL 2	253
	ESC CTRL DELETE	254
	ESC CTRL INSERT	255

WORD FIND continued from page 71

```
1190 REM * PRINT MULTIPLE DIFFERENT CO
PIES ?                  *
1200 IF NCOP>1 AND TYP=1 THEN NCOP=NCO
P-1:CLOSE #5:GOTO 1250
1210 CLOSE #3:CLOSE #5
1220 ? :? "PUZZLES COMPLETE."
1230 ? "BYE ......"
1240 END
1250 IF FLG THEN 1290
1260 ? :? "REWIND CASSETTE AGAIN PLEAS
E"
1270 ? "PRESS PLAY AND THE RETURN KEY"
1280 REM RESTORE CASSETTE WORD FILE
1290 OPEN #5,4,0,FILE$:WCNT=0:WSV$=""
1300 IF FLG THEN 1320
1310 FOR I=1 TO 128:GET #5,A:NEXT I
1320 GOTO 490
1330 REM * PRINT OUT PUZZLE$ *
1340 FOR I=1 TO 10:? #3:NEXT I
1350 REM THIS IS MY FORM FEED FOR A TE
LETYPE,
1360 REM FOR FANFOLD PRINTER PAPER USE
:
1370 REM 1340 ? #3;CHR$(12)
1380 ? #3:? #3:A=INT((PW-10-LEN(TTL$))
/2)
1390 REM CENTER TITLE PLUS 'WORD FIND
' ON 72 COL TTY
1400 FOR I=1 TO A:? #3;" ";:NEXT I
1410 ? #3;"WORD FIND ";TTL$:? #3:? #3
1420 FOR Y=1 TO PSZ
1430 FOR I=1 TO INT((PW-2*PSZ)/2):? #3
;" ";:NEXT I:REM CENTER PUZZLE
1440 FOR X=1 TO PSZ
1450 A=PSZ*(Y-1)+X
1460 ? #3;PUZZLE$(A,A);" ";
1470 NEXT X
1480 ? #3
1490 NEXT Y
1500 ? #3:? #3
1510 RETURN
1520 REM * CREATE NEW WORD FILE FOR WO
RD FIND *
1530 ? "FILE NAME TO PUT NEW"
1540 ? "WORD LIST ON ";:INPUT FILE$
1550 IF LEN(FILE$)=0 THEN ? "NO FILE N
AME ? NO GOOD -":GOTO 1530
1560 FLG=1:IF FILE$(1,1)="C" THEN FLG=
0
1570 IF FLG THEN 1620
1580 ? :? "REWIND BLANK CASSETTE, PRES
S"
1590 ? "PLAY/RECORD AND PRESS THE"
1600 ? "RETURN KEY WHEN CONSOLE SPEAKE
R"
1610 ? "BEEPS TWICE..."
1620 OPEN #2,8,0,FILE$
1630 IF FLG THEN 1650
1640 FOR I=1 TO 128:? #2;"0";:NEXT I
1650 I=1
1660 ? :? I;" WORD = ";:INPUT WORD$
1670 IF LEN(WORD$)>14 THEN ? WORD$;" T
OO LONG":GOTO 1660
1680 IF LEN(WORD$)=0 THEN ? "NO WORD??
'END' IF DONE...":GOTO 1660
1690 ? #2;WORD$
1700 I=I+1:IF I>3*PSZ THEN GOTO 1740
1710 IF WORD$<>"END" THEN 1660
1720 CLOSE #2
1730 RETURN
1740 ? :? "MAXIMUM WORDS ALLOWED REACH
ED"
1750 ? "FINISHING UP THE FILE..."
1760 ? #2;"END":? "END"
1770 CLOSE #2
1780 ? "CONTINUE..."
1790 RETURN
1800 ERR=PEEK(195):LINE=PEEK(186)+256*
PEEK(187)
1810 ? :? "* ERROR # ";ERR;" AT LINE "
;LINE;" *"
1820 ? :? :? "CHECK YOUR ERROR CODE AN
D FIX IT"
1830 END
```

TYPO TABLE

Variable checksum = 885431

Line num range	Code	Length
10 – 90	RN	511
100 – 210	YE	460
220 – 330	JK	510
340 – 470	ON	546
480 – 590	JB	444
600 – 710	UH	509
720 – 830	QY	412
840 – 950	WA	315
960 – 1070	YM	260
1080 – 1190	BX	512
1200 – 1310	YF	366
1320 – 1430	ZW	468
1440 – 1550	TP	280
1560 – 1670	XW	375
1680 – 1790	ZZ	296
1800 – 1830	UO	146

ATTENTION 1200 XL OWNERS

Suffering from the incompatibility blues? Atari has an answer for you. They have just announced the availability of a program on disk, called "Translator," that disables the 1200XL Operating System and loads in a version of hte Operating System used in the Atari 800.

With this disk booted, you can load and run almost all commercial software that previously would not work on the 1200XL, including **Letter Perfect** (LJK) and **The Atari Word Processor** (Atari). The disk is available now from Atari at *no cost*. The toll-free number for Atari from out of California is (800) 538-8543, and from inside California, (800) 672-1404.

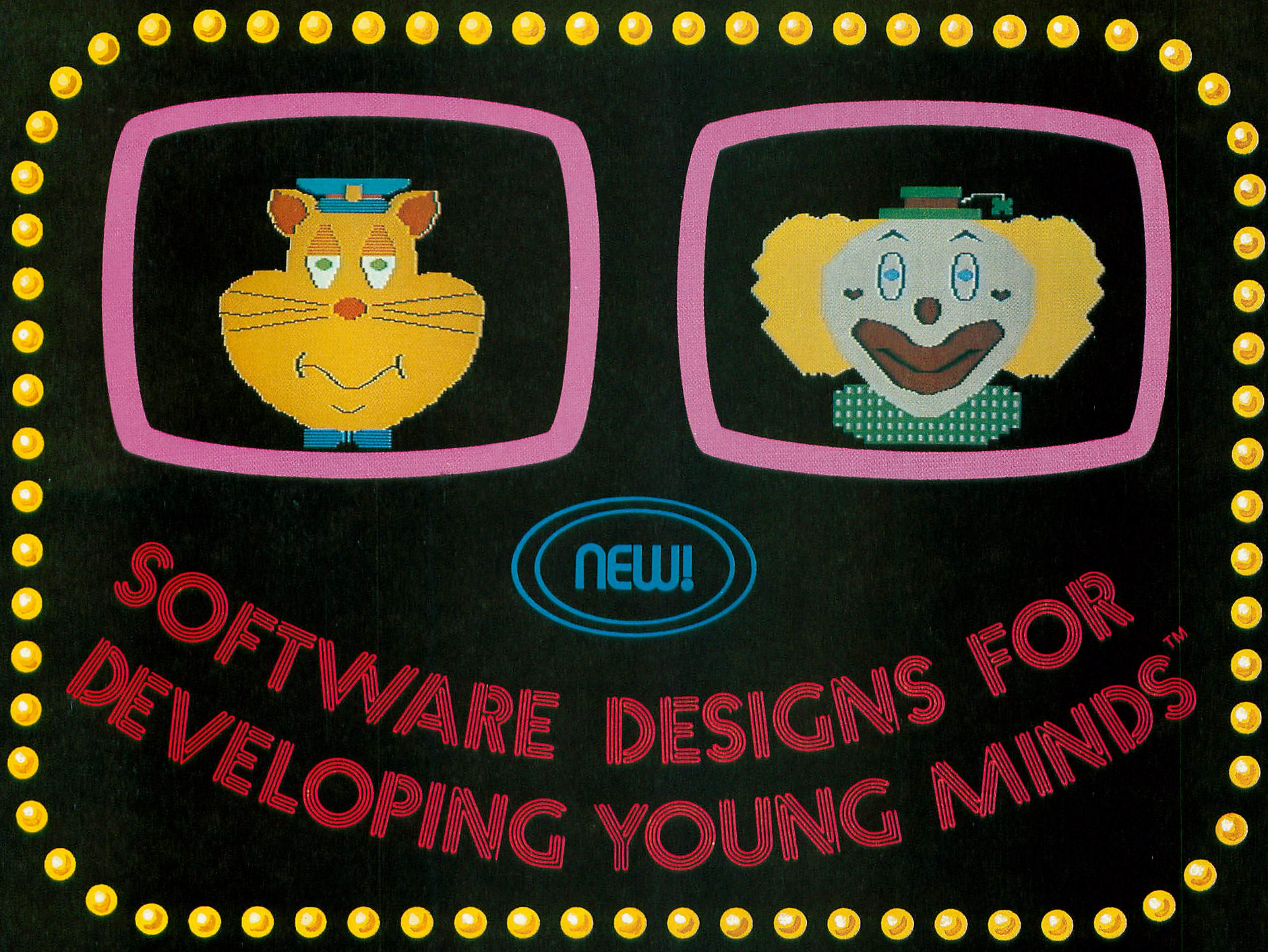
ERRATA

We wish to correct an error which appear in our November *Product Review* section. The address printed for SSI, creator of *Cosmic Balance* was an outdated one. It should have read:

Strategic Simulations, Inc.
883 Stierlin Rd.
Bldg. A-200
Mountain View, CA 94043

Besides the toll-free phone number that we published, they also have a technical hotline, which is operative from 1–5 p.m. (Pacific Time) daily. All questions regarding their products are welcome. The number is (415) 964-1200.

SOFTWARE MOVIES™



MAXIMUS, Inc. is excited to present two new educational software movies for kids of all ages . . . and more are on the way!

STORYLINE™ makes bedtime, or anytime, a warm and friendly funtime. Clover the Clown is your tourguide for two fairytales: The Ugly Duckling and Rumpelstiltskin. Through computer magic you become a part of each story!

SAFETYLINE™ combines fun with important lessons about safety. Max the Cat shows you how to cross the street safely going to school and what to do when lost at the zoo.

Both software movies have interactive games that reinforce the fun and learning. Both have real human voices; you don't need a voice synthesizer!

Features of the software include:

- Sophisticated Face Animation
- Lip-Synchronized Voices
- Original Musical Scores
- Superior Full-Color Graphics
- Optional Joystick Control
- Machine Language Programs

Relax and enjoy yourself. Let Max, Clover, and their colorful friends guide you through a new world of fun and learning. You'll be amazed at how real they are!

STORYLINE™ and SAFETYLINE™ are currently available for any Atari computer with 48K. Each software movie comes in either CASSETTE ONLY ($29.95) or CASSETTE/DISK ($34.95) versions. Order by mail or phone. Visa and Mastercard welcome. Include $2.50 postage and handling; Virginia residents add 4% sales tax. Dealer inquiries invited. Write for our free catalog. MAXIMUS, Inc., 6723 Whittier Avenue, McLean, Virginia 22101.

ORDER TOLL-FREE 1-800-368-2152

MAXIMUS *...Where Excellence is the Standard*

Adventureline™ Series Coming Soon

SHOPPER'S GUIDE

YOUR AD COULD APPEAR HERE CALL 415/661-3400

NEW! UPDATE YOUR ATARI® KEYBOARD WITH

GRAPH-FIX™ KEYBOARD LABELS

- Convenient
- Saves Time
- Fits All Models
- Improves Accuracy
- 29 Peel-and-Stick Symbols to Label each Graphics Key

TO ORDER send $5.95 check or money order (Texas residents add 5% sales tax) to:

DOVESTAR CREATIVE CONCEPTS
P. O. Box 2109, Dept. 1113, Nederland, Texas 77627
(409) 727-5978 (no collect calls, please) or:
MASTERCARD and VISA customers call Toll-Free Monday-Friday 9 a.m. - 5 p.m. Central Time Zone — 1-800-531-7383 (In Texas 1-800-292-5098).
After dialing, get a second dial tone, WAIT for operator and ask for "EXPRESS NUMBER 1037."

DEALER INQUIRIES RECOMMENDED!
Atari TM of Atari, Inc.

GIVE YOURSELF ONE FOR CHRISTMAS!

MEMORY EXPANSION BOARDS for ATARI* COMPUTERS

Christmas Specials

Tiny Tek, Inc. Memory Boards are fully assembled, tested, and guaranteed.

48K/52K Memory Board For ATARI* 400, 52K Addressable Memory, Easy to Install	**$99.95**
32K Memory Board For ATARI* 400 or 800	**$69.95**
16K Memory Board For ATARI* 800	**$49.95**

BUILD YOUR OWN MEMORY

48K/52K Board (No Components)	**$30.00**
32K Board (No Components)	**$25.00**
16K Board (No Components)	**$10.00**
48K/52K Complete Kit	**$85.00**
32K Complete Kit	**$55.00**
16K Complete Kit	**$35.00**

Add $2 Shipping & Handling
Visa & MasterCard Accepted
*ATARI is a trademark of Atari Inc.
Dealer Inquiries Welcome

Tiny Tek, Inc.
P.O. Box 12609 • Dallas, TX 75225
214-373-8926



GO CAMPING WITH ATARI THIS SUMMER AND GET PAID!

ATARI COMPUTER CAMPS needs computer knowledgeable

- Classroom Teachers • On-Site Experts
- Teaching Assistants/Counselors

Send us your resume TODAY!

ATARI COMPUTER CAMPS
Dept AN
1399 Moffett Park Dr
Sunnyvale, CA 94088
Attn. Pat Tubbs

"Beautiful locations all across the Country"

800XL IS A TRADEMARK OF ATARI INC. ©1983 ATARI INC. ALL RIGHTS RESERVED
AN EQUAL OPPORTUNITY EMPLOYER M F H V. A WARNER COMMUNICATIONS CO.

EPSON*, NEC*, PROWRITER*, GEMINI*, OKIDATA 92*

NEW! ATARI* — Allen Macroware PRINTWIZ

The only self-booting grafix handler for dumps in **horizontal** format — all mach. lang. — **Lister** incl. — all modes — mixed modes — change aspect ratios, etc. while running other programs — assem ed — basic or no cartridge — demos, utilities, fonts, included — dump, create forms, stationery, calendars, requires interface. **$26.95**

diskwiz-II

Now for single/double density. Repair, explore, alter, duplicate, map, speedcheck, bad sector (810), block move, trace, special print capabilities, disassembler, new speed, new ease, new functions, special printing functions, excellent repair tool w/instr. — even better than before! The best repair/editor/duplicator at any price — still at the lowest price. (Updates avail. for a small fee.) **$28.95**

1st Class Postage Paid
California Residents add 6%, Foreign Orders add $2.50
C.O.D. add $2.00 — No credit cards
Prices subject to change
(213) 376-4105

ALLEN MACROWARE
P.O. Box 2205
Redondo Beach, CA 90278

*Indicates Trademark of non-related company

MicroMate Software

GRIDIRON GLORY . . . Coach your favorite pro team to victory. 1980, 81, 82 season statistics. 1 or 2 players. Requires BASIC, 48k, Disk Drive . . $17.95*. Trade in your old 32k version for a $10.00 discount.

RAMBLIN GAMBLIN . . . Play Roulette, Black Jack, 5 Card Stud, Slot Machines and Horse Racing as you Ramble and Gamble through the Casino to the Horse Track in this Hi Res board game for 1-4 players, or play any of the games separately. Requires BASIC, 32k, disk drive . . $14.95*

ASTEROID ARTIST . . . Let the kids create a cosmic masterpiece as they pilot their spaceship from star to star, learning numbers and the alphabet as they go. Allows easy coloring in Hi Res Graphics. Requires 32k, disk drive . . $14.95*

10 DAY MONEY BACK GUARANTEE
SEND CHECK OR MONEY ORDER TO
MicroMate Software
Box 20245
Tallahassee, Florida 32316
*ADD $2.00 SHIPPING AND HANDLING

RATED X

XBASIC by George Schwenk

An extension of ATARI® BASIC, **XBASIC** provides 30 new functions, including:

- **Integer and String Arrays**
- **Player Missile Graphics**
- **Special Graphics Modes**
- **Vertical Blank Sound**
- **Requires less than 3 K RAM**
- **Detailed Manual / 6 Examples**

All this power for only **$29.95**, plus $2.00 shipping and handling. Order now from SUPERware, specify disk or cassette, MC, VISA, AMX, check, M.O.

SUPERware
2028 Kingshouse Rd.
Silver Spring, MD 20904
(301) 236-4459

Dealer Inquiries Invited.

NEW! Wizard Ware

EDUCATION • GAMES • APPLICATIONS

ZERO ZONE: Maneuver your ship and try to blast through the force shield of the city of Zone. But watch out for the cities' mountain side cannons which are trying to blast you out of the sky with radioactive missiles. Super arcade sound, graphics and action. 100% machine language. 32K DISK/TAPE.

Price $14.95

COPY PERFECT: **A quality disk reproduction system. Will duplicate entire disk in only 2 passes vs. 3 passes in other systems. Copy Perfect is especially designed to copy autoboot/file oriented disks and damaged sectors. Supports multiple drives. Other special features. 100% machine language 48K DISK.**

Price $14.95

All software has full lifetime warranty against defects.

Send for free software catalog.
To order call or write to:

PH. (919) 851-9314

Wizard Ware
P.O. Box 37093
Raleigh, N.C. 27627

ADVERTISERS

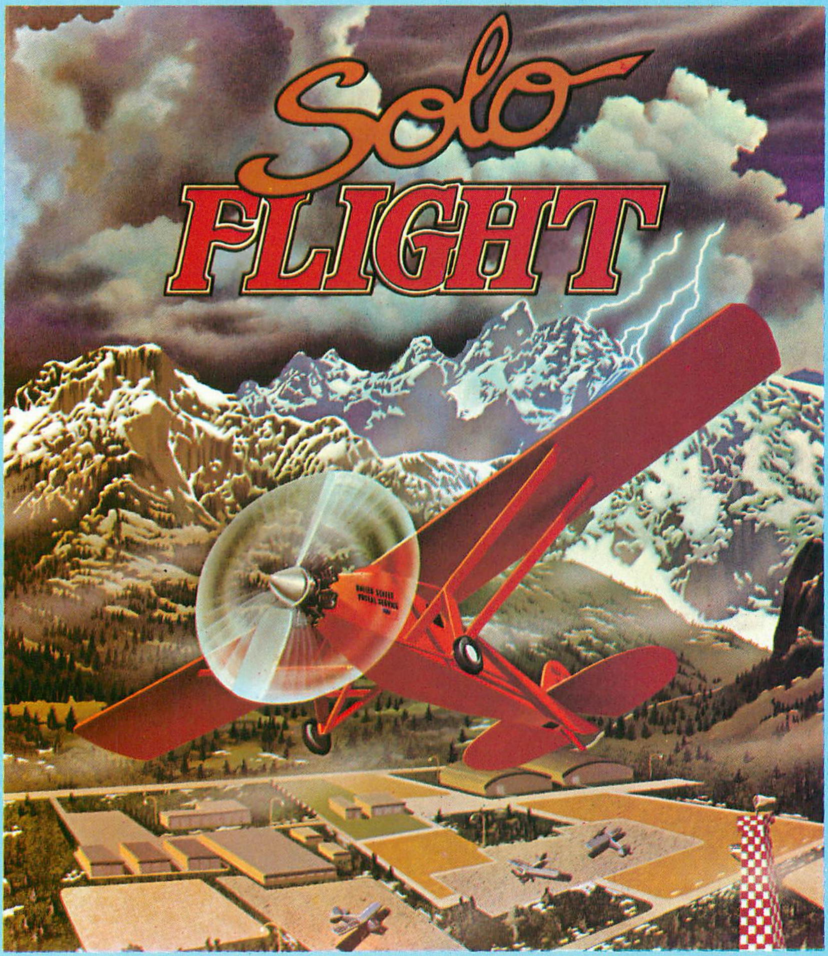
This is provided as a convenience and as a courtesy to advertisers. ANTIC does not guarantee accuracy or comprehensiveness.

AT LAST!!! A REAL FLIGHT SIMULATOR!!!

— From The Creators of HELLCAT ACE!

A real flight simulator allows you the pilot to accomplish Takeoffs, Landings, Cross-Country Navigation, and Emergency Procedure practice under Clear (VFR), Cloudy (IFR), and Cross-Wind conditions. SOLO FLIGHT is such an advanced simulation that uses realistic three dimensional terrain graphics, actual configuration instrument panel, multiple air navigation maps, and accurate aircraft performance characteristics to provide the challenge, thrill, excitement, and joy of the flight experience!!!

- Extensive flight manual including Basic Flying, Instrument Flying, Maps, and Charts.
- Three geographical air navigation areas each with multiple airports and multiple runways.
- Dual Radio Navigation (VOR) and Instrument Landing System (ILS).
- On-screen flight path review between flights.
- Multiple Landing Approach Charts and Design-your-own approach feature.



- Takeoffs and Landings under all weather conditions.
- Cross-Country Navigation via Landmarks, VOR Stations and Dead Reckoning.
- Emergency Procedure Practice.
- Air Mail delivery game to test your flying skills.
- Landing ratings for smoothness and speed.
- For all Atari computers, Commodore 64, Disk and Cassette . . . $34.95.





Look for MicroProse's two new exciting real time combat simulations — MIG ALLEY ACE & NATO COMMANDER. Write or call for our Free Catalog.

If you cannot find our games at your local store, you can order by MasterCard or VISA, Money Order, COD or Check. Add $2.50 for Postage and Handling. Maryland Residents add 5% Sales Tax.

MicroProse Software

DEALER INQUIRIES INVITED.

10616 Beaver Dam Road, Hunt Valley, MD 21030
(301) 667-1151


SELF TEST
MEMORY
AUDIO-VISUAL
KEYBOARD
ALL TESTS
SELECT, START OR RESET
AMDEK
ATARI 1200XL